Semantic Analysis of the Driving Environment in Urban Scenarios

Fahad Lateef

UBFC
UNIVERSITÉ
BOURGOGNE FRANCHE-COMTÉ

utbm
université de technologie
Belfort-Montbéliard

THÈSE DE DOCTORAT

DE L'ÉTABLISSEMENT UNIVERSITÉ BOURGOGNE FRANCHE-COMTÉ

PRÉPARÉE À L'UNIVERSITÉ DE TECHNOLOGIE DE BELFORT-MONTBÉLIARD

École doctorale n°37

Sciences Pour l'Ingénieur et Microtechniques

Doctorat d'Genie Informatique, Automatique et Traitement du Signal

par

FAHAD LATEEF

Semantic Analysis of the Driving Environment in Urban Scenarios

Thèse présentée et soutenue à Montbeliard, le 15 December 2021

Composition du Jury :

BURIE JEAN-CHRISTOPHE	Professeur à l'Université de La Rochelle	Président
TABIA HEDI	Professeur à l'Université d'Evry	Rapporteur
AINOUZ SAMIA	Professeure INSA Rouen	Rapporteur
DAVOINE FRANCK	Professeur à l'Université de INS2I	Examinateur
LI YOU	ingénieur de recherche chez Renault	Examinateur
RUICHEK YASSINE	Professeur à l'Université de Technologie de Belfort-Montbéliard	Directeur de thèse

N° | X | X | X |

ACKNOWLEDGMENTS

First of all, I am deeply indebted to Prof. Dr. Yassine RUICHEK for allowing me to work under his supervision at UTBM, for his patient help and wise guidance. I learned a lot from him, not only about the technical insights and expertise, but also about the way of scientific thinking.

I would like to acknowledge my thesis committee, Prof. Dr. Jean-Christophe BURIE, Prof. Dr. Hedi TABIA, Prof. Dr. Samia ANIOUZ, , Prof. Dr. Frank DAVOINE, and Engr. Dr. LI You for accepting and evaluating my thesis work.

A special thanks to all my colleagues and members of the CIAD lab who have encouraged my research and allowed me to grow as a researcher. It has been a great pleasure to work Dr. Zhi YAN, Dr. Nathan CROMBEZ, and Dr. Jocelyn BUISSON over the past years. My special thanks go to Mohamed KAS for contributing to several papers and for many useful discussions and insightful comments.

I would like to express my gratitude to my father, Prof. Abdul LATEEF, and my mother for their unwavering support throughout my doctoral studies. My wife, Pulwasha and my sons, MUHAMMAD Yousaf and MUHAMMAD Owais, their love are the driving force deep in my heart and have encouraged me not to give up. I am also grateful to my brothers, Umer LATEEF, Abdul AHAD, Abdul WAHAB and Abdul RUB for supporting me. Last but not least, my appreciation goes out to my friends, Mehrose IQBAL, Abderrazak CHAHI, Abdellatif El-Idrissi, Ibrahim KAJO, Ihsanne HOUHOU for their encouragement and support all through my studies.

I dedicate this work to my sister

ATIYA

Miss You

CONTENTS

I

Context and Problem

1

INTRODUCTION

1.1/ AUTONOMOUS DRIVING

The concept of autonomous driving has been around for nearly a hundred years, but the first self-sufficient and truly autonomous cars appeared in the 1980s Fenton (1970) Dickmanns (2002). Since then, various automakers, including General Motors, Mercedes-Benz, Tesla, Toyota, Ford, Audi, Nissan, have developed working autonomous vehicles. Research institutions and tech giants such as Google, Waymo, NVIDIA, Uber, Autonomous Vision Group of the University of Tuebingen, Daimler AG, Max Planck Institute for Informatics, VisLab of the University of Parma, Visual Inference Lab TU Darmstadt, and many others are seriously engaged in autonomous driving. The Society of Automotive Engineers (SAE) has published the international standard J3016 international (2016) "Levels of Driving Automation" for consumers. This sets out six levels of driving automation, from SAE Level Zero (no automation) to SAE Level 5 (full autonomy), summarized in **Figure 1.1**.

- Level 0 (*No Automation*): The human driver is fully in control of the vehicle all the time.
- Level 1 (*Driver Assistance*): The human driver is still in control with few functions carried out by the vehicle, either lane-centering or adaptive cruise control.
- Level 2 (*Partial Automation*): The driver is still committed to the driving, though adaptive cruise control and lane-centering functions are taken simultaneously by the vehicle.
- Level 3 (*Conditional Automation*): All driving functions can be fully undertaken by the vehicle under limited conditions and will not operate unless all the required conditions are met. Human intervention is still required when requested. Traffic jams, chauffeur are example features.
- Level 4 (*High Automation*): The vehicle is full in-charge to perform all driving functions without human intervention but under limited conditions.
- Level 5 (*Full Automation*): The vehicle perform all driving functions in any conditions.

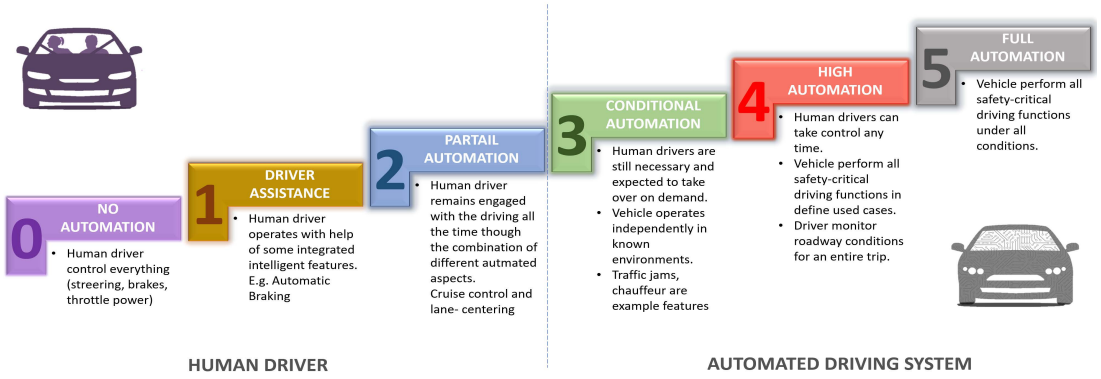


Figure 1.1: The levels of autonomous driving.

The benefits of automated vehicles are mainly in the areas of safety (comfortable driving, potential to eliminate human error, fewer accidents, lives saved, fewer injuries), efficiency, and convenience (smoother traffic flow, less congestion, stable speed profiles, and smoother driving). Also, environmental improvements (energy consumption), mobility (beneficial for the disabled, elderly, underage travelers, and those who cannot afford to own a car), increased capacity (better coordination between road users and reduced safety gaps), reduced transportation costs (in terms of time and stress), and many social and economic benefits.

1.1.1/ CHALLENGES

There are several difficulties towards full autonomy for future autonomous vehicles, and to figure them out we need to understand the autonomous driving system. The system consists of three main parts (perception, planning, and control) shown in **Figure 1.2**, and each part includes different tasks that are expected to be fully understood by the system. In the perception part, knowledge about the vehicle environment (including road, traffic, vehicle location, and obstacle information) is perceived by various sensors such as camera, lidar, radar, GPS, and inertial sensors. The field of computer vision includes methods for analyzing raw sensor data and processing them into meaningful structured information for understanding the environment. These methods work with the various input data from different sensors: for example, object detection and tracking might input data from a camera, LIDAR, and RADAR; traffic light detection, traffic sign classification, and lane detection input data from a camera; localization and mapping system might input data from a camera, LIDAR, and GPS. The planning part uses the results of perception and predicts the intentions of other road elements, i.e., future trajectories: based on the appropriately chosen ego trajectory, driving behavior is created and planned, deciding what explicit action the vehicle needs to take next, what is helpful in high-level route planning for the vehicle, etc. The control part is deeply coupled with the perception and

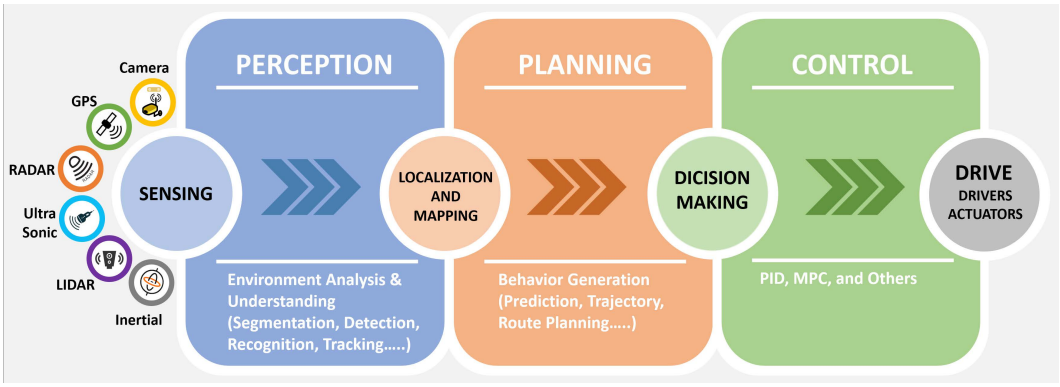


Figure 1.2: Full Autonomous Driving System

planning part. It guarantees that the vehicle follows the course set by the planning part and controls the vehicle's hardware (acceleration, braking, and steering using drivers and actuators) for safe driving.

The companies and researchers have been working very hard to achieve the ultimate goal of Level 5 in autonomous vehicle operation. Today, autonomous technology has reached Level 4 automation, where a vehicle can handle the majority of driving situations independently. However, they still struggle to handle complex traffic situations due to their inability to accurately perceive their surroundings. Perception errors are sure to lead to erratic behavior - and accidents: False or missed object detection, classification or tracking errors, incorrect prediction of movement, unreliable assessment of collision risk, incorrect interpretation of the scene. Therefore, perceptual errors can have potentially catastrophic consequences. Human behavioral variations and unpredictability are major challenges in planning, especially when traffic rules are not followed. According to Rasmussen (1983), there are three types of human behavior: skill-based behavior (activities that occur without conscious attention or control), rule-based behavior (activities that follow a memorized rule or procedure, often based on instructions or preparation), and knowledge-based behavior (activities during an unfamiliar situation that are achieved through previous similar experiences and the combination of rule-based or skill-based behavior). Nevertheless, predicting the behavior of other road elements is also essential for decision-making (control part) in autonomous driving.

Considering that today's challenges in autonomous driving are mainly in the perception part (scene understanding), this requires immense robustness to handle highly complex driving environments. The perception part relies heavily on an extensive infrastructure of active and passive sensors. Active sensors such as LIDAR (create 3D representations of the environment), RADAR (dynamic object detection), GPS, and IMUs for accurate positioning provide an amorphous 3D (geometry) route for the planning part. Besides, the planning part needs semantic information such as the type of objects (e.g., vehicles, pedestrians) to consider their typical dynamics (e.g., speed, direction, position), the state

of regulatory traffic elements (e.g., traffic signs and signals) to comply with traffic rules (e.g., speed limits or stopping), etc. Cameras and computer vision algorithms extract all this semantic information by performing various auxiliary tasks (explained in section 1.1.2) to interpret traffic scenes. Considerable progress has been made in improving perception to achieve scene understanding using Deep Learning-based technology (Goodfellow et al. (2016b)), i.e., equipping machines with a semantic understanding of the world to reliably identify objects and make predictions and actions. However, deep learning models bring with them the well-known shortcomings associated with these trained architectures. Also, interpreting traffic scenarios using computer vision algorithms is far more challenging and complex. Mainly, in urban areas where different road users, static and moving objects may be present, the geometric layout of roads and intersections is variable. Lighting conditions such as cast shadows from vegetation or infrastructure easily confound these image processing algorithms. Also, the limited aperture angle of on-board cameras, their low mounting point, and the limited depth perception of stereo complicate the inference problem, resulting in reliable localization of only nearby objects.

Many factors, summarized in **Figure 1.3**, are stocking to understandability problems for autonomous driving systems from a deep learning perspective. The researchers in the field do not fully understand the dataset, the trained model, and the learning phase. A finite training dataset cannot exhaustively cover all possible driving situations, and it is likely to under- and over-represent some specific situations. The trained model (aspects of generalizability and robustness) and the mapping function it represents are poorly understood and considered as a black box. The model is highly nonlinear and offers no guarantee of robustness, as small input changes can dramatically alter the output behavior. The learning phase is not perfectly understood. Among other things, there are no guarantees that the model will settle at a minimal point that generalizes well to new situ-

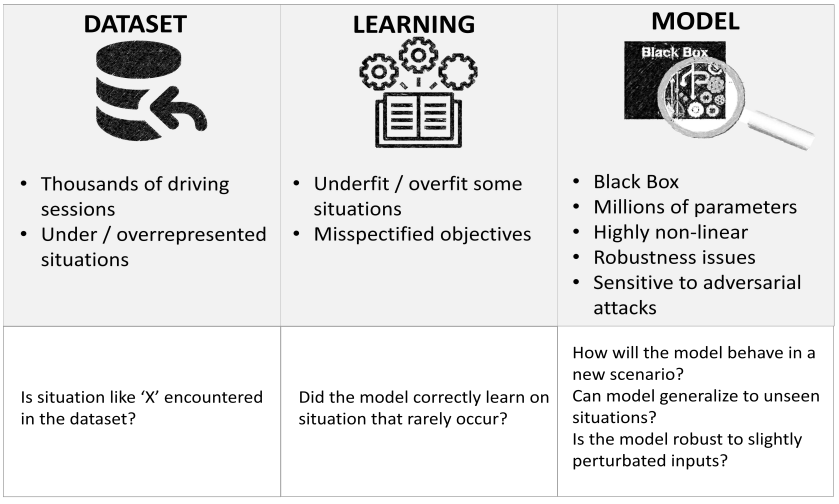


Figure 1.3: Challenges and Questions in Deep Leaning

ations and that the model will not be under-fitted in some situations and overfit in others. Also, during training, the model may learn to base its decisions on spurious correlations rather than using causal signals Zablocki et al. (2021).

1.1.2/ SEMANTIC ENVIRONMENT UNDERSTANDING

Scene Understanding can be viewed as the process of adding and extracting semantic information from the sensor data characterizing a scene, or scene understanding is the analysis of a scene, taking into account the semantic and geometric context of its contents and the internal relations between them. We humans can understand a complex dynamic scene only from its projection into our eye by classifying, locating, segmenting, and identifying objects and features at one look. These tasks are performed sequentially to form a consistent process that provides an output of valuable information for semantic understanding of the projected image. **Figure 1.4 (a) Raw images** can be defined as images of outdoor scenes, images of urban driving scenes, or scene images with multiple dynamic (vehicles, pedestrians, cyclists, and tram) and static (buildings, sky, road, and trees) objects. Humans have the ability to classify (object type and status moving/static), specify (spatial position), identify (motion, position, direction, and velocity), and track these objects in the driving scene. In addition, humans focus their visual attention more on important or purposeful elements and ignore unnecessary ones in their field of view. These properties are usually interrelated, and humans can easily associate them with the scene at different levels. Conferring these phenomenal abilities into machine-learning systems has been a long-standing goal in the field of computer vision.

Numerous approaches and methods have been proposed to improve scene understanding and extract semantic information about the driving environment from images and videos. Deep learning (**DL**) Goodfellow et al. (2016b) is now ubiquitous in computer vision, which has adopted deep convolutional neural networks to understand high-dimensional data, such as images and videos. Representations are learned by encoding inputs through multiple nonlinear layers and sub-sampling operations, resulting in strong image-level understanding and recognition capabilities. Thanks to significant technological advances at both the hardware (computational speed) and software (strong and robust, using multiple neural networks) levels, the methods of DL have achieved amazing results. They have been mainly used in computer vision for image recognition tasks Krizhevsky et al. (2012) He et al. (2016). Since then, one has observed the adaptation of DL methods in various computer vision and image-based scene understanding tasks, such as object detection Mottaghi et al. (2014), semantic segmentation Kemker et al. (2018), motion estimation/compensation Yu et al. (2019b), depth estimation Jiang et al. (2017), saliency prediction Borji (2019), etc. These tasks can typically be formulated as image labeling problems, where labels are assigned to a set of locations corresponding

to image pixels. They differ in the requirements for human supervision and the amount of work required to generate the labels.

In recent years, much research has been done in the field of " ***Object Detection*** ", where the goal is to localize objects with a bounding box and object types or classes in an image. Object detection algorithms restrict the semantic information to different categories, e.g., building, road, sky, trees, and sidewalk are considered in one background category, while the rest of the objects are in different categories. Object detection approaches are very efficient, especially for frequent occurrences of objects such as cars Li et al. (2020c), persons Dollar et al. (2011) due to a great number of training samples and comparatively low intra-class variance. A much stronger representation is achieved by the task " ***Semantic Segmentation*** ", which assigns a semantic category or class such as car, pedestrian, building, road, sky, etc. to each pixel in an image. It estimates the probability that the pixel belongs to a set of the defined object class. Several methods for semantic segmentation have been developed in the community and have made important contributions to the field Feng et al. (2020), Lateef and Ruichek (2019). Other commonly studied tasks for scene understanding problems include " ***Optical-Flow*** & ***Ego-Motion*** " and "***Depth*** & ***Shape Estimation*** ", which represent different aspects of objects in a scene, i.e. object motion and geometry. Optical flow encodes temporal-visual information from image sequences and is often used to relate scene changes over time. Ego-motion is defined as the three-dimensional movement of a camera within an environment. Depth estimation refers to algorithms that aim to obtain a representation of the spatial structure of a scene. Each of these tasks provides different cues to understanding the scene and could be correlated. The motion of the object (flow and semantic) provides specific cues to its motion pattern, and the geometry of the object provides cues to depth and shape. Several works, outlined in the literature, estimate optical flow and depth information from stereo image pairs or video sequences Ilg et al. (2017), Rateke and von Wangenheim (2020). Another important task that has been explored for scene understanding is "***Saliency Prediction for Visual Attention***," where the goal is to detect salient regions that correspond to important objects and events in a scene and their mutual relationships. This ability is fundamental to the way humans perceive and interpret a scene. Their visual system selectively focuses its attention on salient parts and performs a detailed understanding of the most salient regions.

This work aims to emulate some of the utilities of human behavior and build image representations that can efficiently facilitate semantic information associated with the image, given some training examples of previously seen semantic concepts. Our goal is to obtain a representation that can be effectively used in applications such as autonomous driving Geiger et al. (2012) and robot navigation Kümmerle et al. (2015).

1.2/ OBJECTIVES AND CONTRIBUTIONS

This thesis investigates, designs, implements, and evaluates classical and deep learning-based solutions for semantic analysis of the driving environment in urban scenarios, where we mainly deal with image and video processing. We restrict ourselves to the area of scene perception (understanding scene semantics and visual attention) and ways to improve it using semantic segmentation, motion estimation, depth estimation, saliency prediction for visual attention, as well as other available cues. We must consider the difficulty of having real ground truth data available to train supervised models for these tasks. In this work, cameras are intended to be the primary control and do not cover additional sensory information (LIDAR, radar, IMU, GPS...). We seek to provide an autonomous vehicle moving in an urban environment to adequately perceive, analyze, and interpret traffic as humans do.

Objectives of this thesis are as follows:

1. Give a comprehensive overview of deep learning techniques used for semantic segmentation, which is the most studied topic in the literature for understanding urban scenes.
2. Understand the geometric structure of the urban driving scene and the Spatio-temporal evolution of the participants (vehicle, pedestrian, cyclist, etc.). The ultimate goal is to semantically reason about the scene's evolution to provide clues that can aid in decision making and autonomous vehicle control.
3. Understanding the processes that determine where one looks in scenes (Visual Attention) is one of the fundamental goals in the study of scene perception. The third objective is to investigate known saliency algorithms (classical and deep learning approaches) for their applicability in visual saliency for multiple objects in driving scenes.
4. Propose a DL-based solution for visual attention that highlights road context objects as salient in driving scenes. Furthermore, we seek to ensure consistent robustness (generalization performance) and high accuracy of the solution under various adverse conditions.

We would like to point out that all of the above objectives were not necessarily implemented in the order in which they were described, but we present them in the following order for the sake of comprehensibility. The work formulates classical and deep learning methods for several vision tasks' strengths to achieve better semantics and visual cues for understanding driving scenes in cities. These tasks include many processing possibilities, e.g., semantic segmentation, instance segmentation, moving object detection,

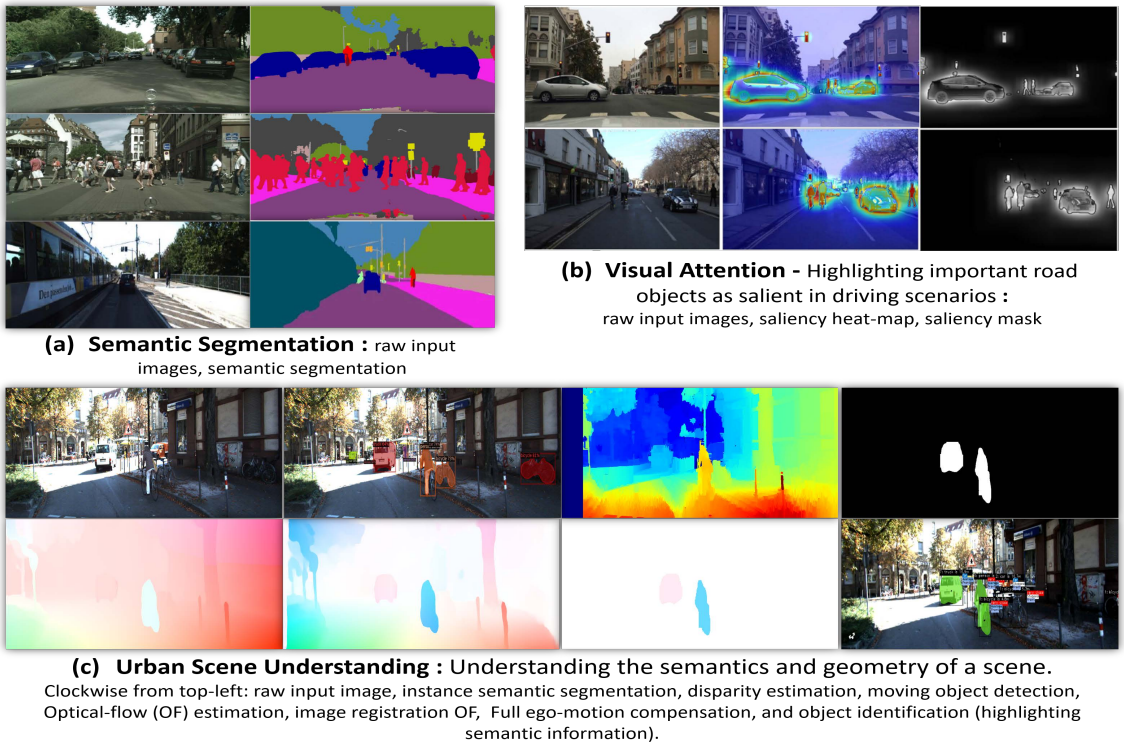
motion compensation and estimation, stereo disparity/depth estimation, and saliency prediction for visual attention. The main contributions are

- ➔ We begin by studying the advances and innovations in Deep Learning and semantic segmentation. There is a dearth of state-of-the-art reviews on these topics. Deep Learning is a new sub-field of machine learning that is growing at a rate that makes it very difficult to stay up to date, even following the work that is being done in semantic segmentation. This includes developing new methods, improving existing methods, and using them in new application domains. Therefore, we first created a taxonomy to classify these methods and approaches into ten different classes based on the common concepts of their architectures. We review the state- of- the-art techniques and analyze their architectures to find out how they achieve their stated performances. We provide a detailed survey of publicly available datasets on which these methods have been evaluated. We also point out some open problems in semantic segmentation and their possible solutions.

- ➔ Next, we developed a new framework for visual attention in driving scenarios highlighting objects in the road context as salient based on Generative Adversarial Network. We started with a review of well-known saliency algorithms, including classical and deep learning approaches, used for visual attention and tested these algorithms for their applicability to visual saliency for multiple objects in driving scenes. We add a new scheme to generate data for a model of visual attention in autonomous driving. An extensive Visual Attention Driving Database (VADD) of heatmap labels is created from publicly available driving nature datasets that contain various driving activities and environments, including rain, night, snow, highways, and urban scenes.

- ➔ In the next step, we seek to extend our visual scene understanding solution by incorporating motion and distance information about the various components of the driving scene. We have developed a framework that can detect objects and extract their behavioral characteristics in terms of motion, position, velocity, and distance to better understand the driving scene. We design a moving object detection model within the framework by integrating an encoder-decoder network with a segmentation model. The approach involves two mutual tasks: Object segmentation of specific classes and binary pixel classification to predict whether the detected object is moving or static based on temporal information. We propose to use image registration as a tool to compensate for ego-motion due to the moving camera and then compute optical flow to extract the actual motion information of the moving objects. The work contributes a novel dataset for moving object detection that covers all kinds of dynamic objects.

The work advances state-of-the-art tasks with effective and efficient models, and outperforms previously published approaches on some of the problems mentioned before. Examples of the variety of methods developed and used in this thesis are shown in **Figure 1.4**.



(a) Semantic Segmentation : raw input images, semantic segmentation

(b) Visual Attention - Highlighting important road objects as salient in driving scenarios : raw input images, saliency heat-map, saliency mask

(c) Urban Scene Understanding : Understanding the semantics and geometry of a scene. Clockwise from top-left: raw input image, instance semantic segmentation, disparity estimation, moving object detection, Optical-flow (OF) estimation, image registration OF, Full ego-motion compensation, and object identification (highlighting semantic information).

Figure 1.4: Examples of the variety of methods developed and used in this thesis. (a) Semantic Segmentation (b) Visual attention for urban driving and (c) Understanding the semantics and geometry of a scene.

1.3/ THESIS OUTLINE

The main body of this thesis is divided into three chapters, each containing one or more contributions. The chapters address core computer vision tasks for scene perception: a taxonomy of deep neural network-based semantic segmentation approaches is given in Chapter 2, visual attention for urban driving in Chapter 3, and disparity estimation, moving object detection, and motion compensation/estimation for urban driving scenes in Chapter 4. For each chapter topic, the state of the art is discussed. We present formulations for deep learning architectures and discuss how they can be used to improve results for all the tasks considered. In Chapter 5, we draw general conclusions, and suggest directions for future research.

II

CONTRIBUTION

2

Deep Semantic Segmentation Taxonomy

2.1/ Introduction

In this chapter we give a comprehensive overview of semantic segmentation using the methods of Deep Learning. We have classified these methods into ten classes, according to the common concepts underlying their architectures. The categories are presented in tabular form, with each method, its main idea, the origin of its architecture, test benchmarks, and code availability. This categorization provides a complete summary of the methods, which both inspire and diverge from each other. The chapter also gives an overview of the publicly available datasets on which the studied methods have been evaluated. It also presents the evaluation matrices that were used to measure their accuracy. A detailed analysis of the known methods and their architectures is presented to find out how they achieve their stated performances. Later, the open problems and their possible solutions are discussed.

2.2/ Semantic Segmentation

Semantic segmentation is the most studied research topic and core task in scene understanding. This task relates to the labeling of each pixel in an image with its corresponding semantically meaningful category. Recent work in Deep Learning dealing with semantic segmentation has been greatly enhanced by the use of neural networks. Neutral networks have made tremendous strides with the availability of large amounts of data thanks to the advent of digital cameras, cell phone cameras, and the ever-faster processing power of GPUs. Semantic segmentation has several applications in computer vision & artificial intelligence - autonomous driving Feng et al. (2020), robot navigation Zhang et al. (2018b), industrial inspection Tao et al. (2018); remote sensing Kemker et al.

(2018); in cognitive and computer sciences - saliency object detection Luo et al. (2021); in agricultural sciences Milioto et al. (2018); fashion - clothing categorization Martinsson and Mogren (2019); in medical sciences - medical image analysis Taghanaki et al. (2021) etc. The earlier approaches used for semantic segmentation were texton forest Shotton et al. (2008), random forest based classifiers Shotton et al. (2011a), while deep learning techniques provide accurate and much faster segmentation.

2.2.1/ Review - Deep Learning Architectures

The first successful application of convolutional neural networks was developed by LeCun et al. (1998). They presented an architecture called LeNet5 to read zip codes and digits and extract features at multiple locations in the image. Later, Krizhevsky et al. (2012) published a large Deep Convolutional Neural Network (AlexNet) , which is considered one of the most influential publications in the field. AlexNet is a deeper and wider version of LeNet used for learning complex objects and object hierarchies. Zeiler and Fergus (2014) introduced ZFNet, which is a fine-tuning of the AlexNet structure. They proposed a technique for visualizing feature maps at each layer of the network model. This technique uses a multilayer deconvolutional network to project feature activations back into the input pixel space. Lin et al. (2013) proposed a model Network-In-Network (NIN), based on a multilayer perceptron (MLP) Rosenblatt (1961), consisting of several fully connected layers with nonlinear activation functions. Szegedy et al. (2015) developed an efficient deep neural network called GoogleNet. They introduced an inception module as shown in **Figure 2.1**, which is a combination of 1×1, 3×3 and 5×5 convolutional filters and a pooling layer. It reduces the number of features and operations on each layer, save time and computational cost. Later, Ioffe and Szegedy (2015) proposed an algorithm called BN-Inception for constructing, training and performing inference using Batch Normalization method. Szegedy et al. (2016) introduced two new modules, Inception V2 and Inception V3, making some significant changes (e.g., factorization of convolutions and use of grid reduction methods) to their previous module. In Szegedy et al. (2017), they replaced the filter concatenation stage of the Inception architecture with residual connections to

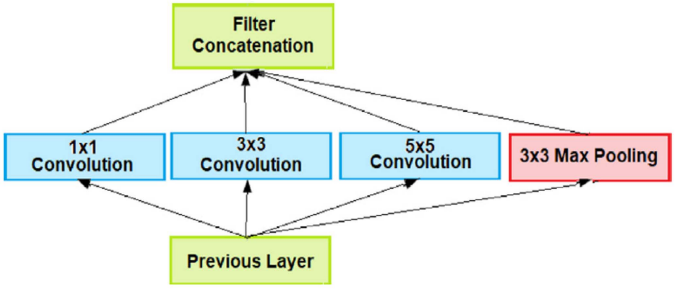


Figure 2.1: Inception module

Table 2.1: **GoogLeNet Modules**

Model	Corpus	Original Architecture	Testing Benchmark	Code Available
GoogLeNet	Inception module: Bottleneck Szegedy et al. (2015)	NIN	ImageNet	YES
	Batch Normalization Modified BN-Inception Ioffe and Szegedy (2015)	Inception	ImageNet	YES
	Inception V2, V3 Szegedy et al. (2016)	BN-Inception	ImageNet	YES
	Inception V4 and Inception-ResNet-v1, 2 Combining the Inception architecture with Residual connections Szegedy et al. (2017)	Inception V3 ResNet	ImageNet	YES
	Xception Chollet (2017) Depthwise Separable Convolutions Mamalet and Garcia (2012)	Inception V3 ResNet	ImageNet JFT (Google's) FastEval14k	YES

increase efficiency and performance. They proposed Inception-ResNet-v1, Inception-ResNet-v2, and an Inception-only variant called Inception V4. Chollet (2017) proposed a module called Xception, which means extreme Inception. They replaced the Inception modules with depth-wise separable convolutions proposed in Mamalet and Garcia (2012). **Table 2.1** shows the GoogLeNet modules.

2.2.1.1/ Feature Encoder Based Methods

The dominant approaches to feature extraction method in the literature are Visual Geometry Group (**VGG**) Simonyan and Zisserman (2014) network and Residual Learning Frameworks (methods that use residual block He et al. (2016) as a fundamental building block in their architecture). In this category, we present these methods and their invariants presented in **Table 2.2**. The idea behind the concept is to extract feature maps based on stacked convolutional layers, ReLu layers and pooling layers.

The VGG network was introduced by the prestigious Visual Geometry Group at Oxford. Unlike LeNet and AlexNet, VGGNet uses multiple 3×3 convolutions in the sequence, which can mimic the effect of larger receptive fields, e.g., 5×5 and 7×7. However, it requires a large number of parameters and high learning power since it uses large classifiers. **Figure 2.2** shows a VGGNet with 16 convolutional layers. Residual Network (**ResNet**) is the most popular and widely used neural network for semantic segmentation. It is difficult to train a deep neural network with a large number of layers. As the depth increases, the performance becomes saturated or even starts to degrade due to the vanishing gradient problem. Several solutions were proposed by Hinton et al. (2006) Hinton (2009) Byeon et al. (2015), but none of them seemed to really tackle the problem. He

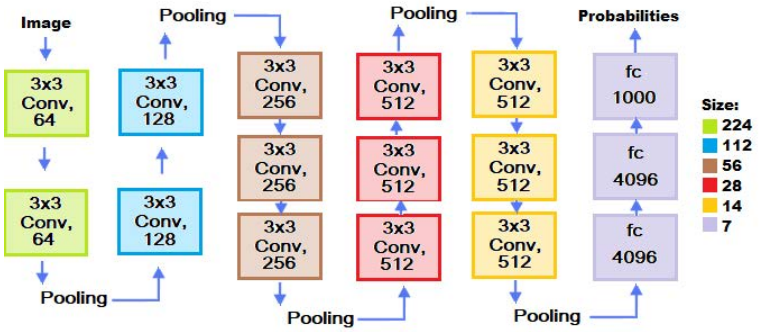


Figure 2.2: VGG-16 Layer Structure

et al. (2016) effectively solved the vanishing gradient problem by introducing an identity shortcut connection (i.e., skipping one or more layers) as shown in **Figure 2.3**. They proposed a residual block with pre-activation variant, where gradients can flow easily and unobstructed through the shortcut connection during back-propagation.

Several architectures are based on ResNet, its variants and interpretations. Paszke et al. (2016) presented an encoder/decoder scheme network called an efficient neural network (ENet). This network is similar to the bottleneck approach of ResNet and is specifically designed for tasks that require low latency, such as mobile phones or battery-powered devices. Pohlen et al. (2017) proposed a Full-Resolution Residual Network (FRRN) with strong localization and recognition performance for semantic segmentation. FRRN exhibits the same superior training properties as ResNet and has two processing streams: residual and pooling. The residual stream carries information at full image resolution and enables precise segment boundary compliance. The pooling stream goes through a sequence of pooling operations to obtain robust features for recognition. The two streams are coupled at the full image resolution using residuals to achieve strong recognition and localization performance for semantic segmentation. Wu et al. (2019b) presented a neural network called ResNet-38, where they added and removed layers in residual networks at training/test time. They analyzed the effective depths of residual units and pointed out that ResNet behaves like linear ensembles of shallow networks. Authors in Sun et al. (2019) proposed an HRNet that maintains the high-resolution representations by combining high- and low-resolution convolutions in parallel and repeatedly performing multiscale fusions over parallel convolutions, achieving strong and spatially precise high-resolution representations. Inspired by HRNet, a deep dual resolution network was developed by Hong et al. (2021) to perform real-time semantic segmentation of high-resolution driving images. The proposed DDRNet consists of two parallel deep branches with different resolutions. One branch generates high-resolution feature maps and the other extracts rich contextual information through multiple downsampling operations. They introduce a novel module called Deep Aggregation Pyramid Pooling Module (DAPPM) that greatly increases the receptive fields and extracts contextual information better than the normal

Pyramid Pooling Module. When integrated with low-resolution feature maps, the model leads to a small increase in inference time. Li et al. (2020b) proposed to use video prediction models to apply labels to immediately adjacent frames. They introduced a joint frame label to mitigate the misalignment problem. They also proposed to relax the training with only one label by maximizing the probability of union of class probabilities along the boundary. Tao et al. (2020) propose an efficient hierarchical multi-scale attention mechanism that helps with both class confusion and fine details by allowing the network to learn how best to combine predictions from multiple inference scales.

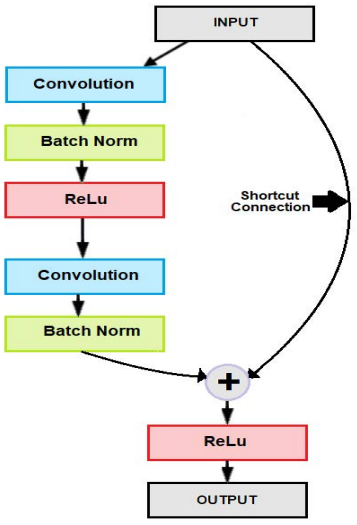


Figure 2.3: Residual Learning: A building block

Adapting the idea of ResNet-50 He et al. (2016), an architecture called Adaptive network or AdapNet is proposed by Valada et al. (2017). They introduced an additional convolution with a kernel size of 3×3 before the first convolutional layer in ResNet, allowing the network to learn more high-resolution features in less time. They also proposed the fusion scheme convoluted mixture of deep experts (CMoDE) to learn robust kernels from complementary modalities and features. The proposed model adaptively weights the class-specific features depending on the scene conditions. Inspired by ENet, Romera et al. (2017) proposed an efficient residual factorized network ERFNet for real-time semantic segmentation. ERFNet proposes a non-bottleneck-1D (non-bt-1D) layer and combines with bottleneck designs in a way that best exploits their learning ability and efficiency. Mehta et al. (2018) developed a convolutional module called efficient spatial pyramid (ESP) for their new efficient neural network. The ESP module consists of pointwise convolutions (reducing complexity) and the spatial pyramid of dilated convolutions (providing a large receptive field). Casanova et al. (2018) presented a Fully Convolutional Dense ResNet, called FC-DRN. The basic idea is to combine the strengths of the network architectures FC-ResNet (gradient flow and iterative refinement) and FC-DenseNet Jégou et al. (2017) (multiscale feature representation and deep supervision).

Table 2.2: **Feature Encoder based Methods**

Category	Strategy / Structure			Main Contribution	Architecture Origin	Testing Benchmark	Code Available
Feature Encoder Concept	**Visual Geometry Group Network** (VGGNet) Simonyan and Zisserman (2014)			Convolutional Networks (ConvNets) Used much smaller 3×3 filters in each convolutional layers which match the effect of larger receptive fields, e.g. 5×5 and 7×7	AlexNet	ImageNet, PASCAL VOC	YES
	FRAMEWORKS	RESIDUAL LEARNING	Residual Network (ResNet) He et al. (2016)	Bottelneck Approach Shortcut Connections are added (MLPs - Multi Layer Perceptrons)	VGG	ImageNet, Cityscapes, CIFAR-10, COCO, PASCAL VOC	YES
			ResNet-38 Wu et al. (2019b)	(Shallow Network) ReNet for Image classification FCN for semantic image segmentation	ResNet + FCN	Cityscapesss, ADE20K, PASCAL VOC	YES
			Fully Convolutional Dense ResNet (FC-DRN) Casanova et al. (2018)	Combining the strength of FC-ResNet: gradient flow and iterative refinement and FC-DenseNet: Multi-Scale feature representation and deep supervision).	ResNet	CamVid	-
			High-Resolution Network (HRNet) Sun et al. (2019)	High-resolution representations by connecting high-to-low resolution convolutions in parallel and repeatedly conducting multi-scale fusions across parallel convolutions.	ResNet	Cityscapes, PASCAL Context, LIP	YES
			Hierarchical Multi-scale Attention Network Tao et al. (2020)	Combine multi-scale predictions together at pixel level. Network learns to predict a relative weighting between adjacent scales.	HRNet	Cityscapes, Mapillary	YES
			Video Propagation and Label Relaxation Li et al. (2020b)	Label Propagation (LP): Pairing a propagated label with the original future frame. Joint image-label Propagation (JP): Pairing a propagated label with the corresponding propagated image.	ResNet	Cityscapes, KITTI, Camvid	YES
			Adaptive Network (AdapNet) Valada et al. (2017)	Convoluted Mixture of Deep Experts (CMoDE) fusion scheme	ResNet	Cityscapes, Synthia, Freiburg forest	-
			AdapNet++ Valada et al. (2019)	Self-Supervised Model Adaptation (SSMA): Fuses modality-specific feature maps based on object class, its spatial location and the scene context.	AdapNet	Cityscapes, Synthia, SUN RGB-D, Freiburg forest, ScanNet	YES
		ENCODER DECODER	Full-resolution Residual Networks (FRRN) Pohlen et al. (2017)	Two Stream Network Residual Stream: Carries information at the full image resolution, enabling precise adherence to segment boundaries. Pooling Stream: Sequence of pooling operations to obtain robust features for recognition	ResNet + VGG	Cityscapes	YES
		REAL TIME	Efficient Neural Network (ENet) Paszke et al. (2016)	Presents a different view on encoder-decoder architecture. The decoder is to upsample the output of the encoder, only to fine-tuning	ResNet	Cityscapes, CamVid, SUN	YES
			Efficient Residual Factorized Network (ERFNet) Romera et al. (2017)	A non-bottleneck-1D (non-bt-1D) layer and combines with bottleneck	ResNet ENet	Cityscapes	YES
			Deep dual-resolution networks (DDRNets) Hong et al. (2021)	Deep Aggregation Pyramid Pooling (DAPPM) module: A combination of deep feature aggregation and pyramid pooling	HRNet	CityScapes, CamVid	YES
			Efficient Spatial Pyramid ESPNet Mehta et al. (2018)	Efficient spatial pyramid (ESP) modules: Spatial pyramid of dilated convolutions	ResNet	CityScapes, PASCAL VOC, Mapillary	YES

The focus on VGG and ResNet approaches in recent work led to remarkable results in semantic segmentation. The residual learning frameworks follow the core idea of ”skip connection”, which is the main intuition behind their success. However, using them on a large scale can lead to memory problems. This pioneering work makes it possible to train deeper networks with good performance.

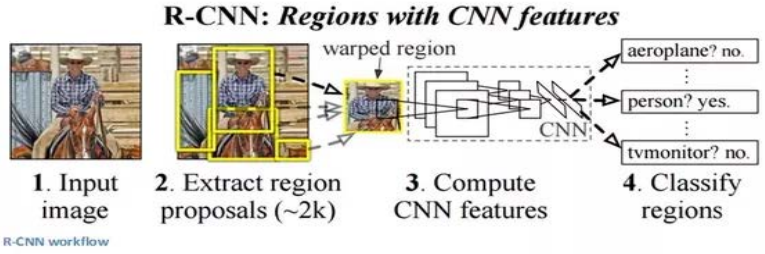


Figure 2.4: The architecture of R-CNN Girshick et al. (2014)

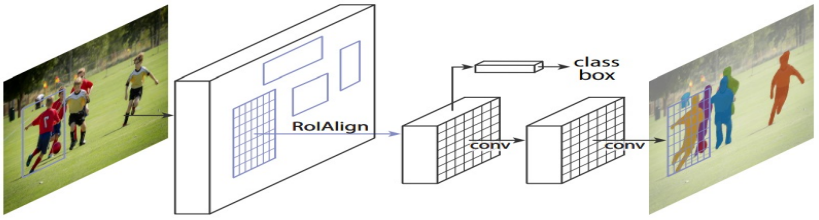


Figure 2.5: The framework of Mask R-CNN He et al. (2017a)

2.2.1.2/ Regional Proposal Based Methods

Regional proposal algorithms are very influential in computer vision (for object detection techniques). The core idea is to detect the regions according to the variety of color spaces and similarity metrics, and then perform classification (region proposals that might contain an object), often called Region-wise prediction. Regional Convolutional Neural Network (R-CNN) along with its derivatives shown in **Table 2.3**.

Girshick et al. (2014) at UC Berkeley proposed a first region-based convolutional neural network (R-CNN) for object detection tasks. The R-CNN consists of three modules: a regional proposal generator, in which the selective search method Uijlings et al. (2013) was used to generate 2000 different regions that have the highest probability of containing an object; a convolutional neural network LeCun et al. (1998) to extract features from each region; finally, these features are used by the CNN as input to a set of class-specific linear SVMs. The features are also fed into the bounding box regressor to obtain the most accurate coordinates and reduce localization errors. **Figure 2.4** shows the R-CNN architecture.

A Fast R-CNN was proposed by Girshick (2015) with a technique called RoIPool (Region of Interest Pooling), which improves training and testing speed and increases object detection accuracy. Later, a team from Microsoft proposed a Faster RCNN architecture Ren et al. (2015). They introduce Region Proposal Network (RPN), which is a kind of fully convolutional network (FCN) built by adding some additional convolutional layers that predefine object boundaries and object hugeness values (set of object classes compared to background) at each position. The RPN generates region proposals (multiple scales

and aspect ratios) that are fed into the Fast R-CNN for object detection. RPN and Fast R-CNN share their convolutional features, which reduces complexity, increases speed, and improves the overall accuracy of object detection. Lin et al. (2017c) introduce Feature Pyramid Networks (FPN), a multiscale pyramid hierarchy of deep convolutional networks (ConvNet's), and create feature pyramids with semantics at all levels that can be used to replace featurized image pyramids with minimal cost (power, speed or memory). He et al. (2017a) proposed a Mask Regional Convolutional Neural Network (Mask-RCNN) that extends Faster R-CNN for pixel-level image segmentation. It added a branch (small FCN) on each RoI for object mask prediction on a pixel-by-pixel basis, in parallel with the existing branch for bounding box recognition (classification and regression). The faster R-CNN has the disadvantage of misalignment (pixel-by-pixel alignment) between network inputs and outputs. Mask-RCNN solves this problem by replacing the RoI pooling layer with Region of Interest Alignment (RoIAlign), a quantization-free layer that maintains exact spatial locations as shown in **Figure 2.5**. Liu et al. (2018) presented a network built on Mask-RCNN and FPN called Path Aggregation Network (PANet), which strengthens the information flow in the context of proposal-based instance segmentation. Recently, Zhang and Chi (2020) considered the advantages of both segmentation and object detection and proposed a network model that combines pixel-based FCN and object-based Mask-RCNN. The network is called Mask-R-FCN and the classification results are fused in the decision level of two (DNNs) for the proposed Mask-R-FCN.

Neural networks based on region proposals have the advantage that object detection

Table 2.3: Region Proposal based Methods

Category	Strategy / Structure	Corpus	Original Architecture	Testing Benchmark	Code Available
Regional Proposals	Regional Convolutional Neural Network (R-CNN) Girshick et al. (2014)	Regional proposal generator: Selective Search Method CNN: for extracting features from each region Set of class specific linear SVMs to score features.	AlexNet VGG-16	PASCAL VOC	YES
	Fast R-CNN Girshick (2015)	Improvement in R-CNN Region of Interest (RoI) pooling layer.	VGG-16	PASCAL VOC	YES
	Faster R-CNN Ren et al. (2015)	Region Proposal Network (RPN) Merge of RPN and Fast R-CNN.	VGG-16 FCN as RPN ZFNet	PASCAL VOC COCO	YES
	Mask R-CNN He et al. (2017a)	Region of Interest Alignment (RoIAlign): for pixel-to-pixel alignment	VGG-16 FCN as RPN ZFNet	Cityscapes, COCO	YES
	Feature Pyramid Network (FPN) Lin et al. (2017c)	Create feature pyramids having semantics at all levels, that can be used to replace featured image pyramids.	Fast/Faster R-CNN	COCO	YES
	Path Aggregation Network (PANet) Liu et al. (2018)	Bottom up Path Augmentation Adaptive Feature Pooling: Fully connected Fusion:	Mask R-CNN / FPN	COCO, Cityscapes, Mapillary vistas	-
	(Mask-R-FCN) Zhang and Chi (2020)	Combining the pixel-based FCN and object based Mask-RCNN	FCN / Mask R-CNN	Zurich GID	-

and segmentation can be achieved simultaneously. The proposals are generated by algorithms (Hosang et al. (2015) provide deep analysis) that are semantically meaningful and related to objects. They may contain an object class or several other classes that can help in determining the semantic labels. Moreover, feeding the wrapped region proposals into a convolutional neural network for classification can reduce the computational cost.

2.2.1.3/ Recurrent Neural Network Based Methods

Recurrent neural networks (RNNs) have actually been introduced for sequence processing Goodfellow et al. (2016b) Graves et al. (2013) Gao et al. (2018). In addition to their success in handwriting and speech recognition, RNNs have been very successful in computer vision (image processing). We have only studied network models that use RNNs in 2D images (integrating convolutional layers with RNNs). The recurrent neural network consists of long Short-Term Memory (LSTM) Hochreiter and Schmidhuber (1997) blocks. The ability of RNN to learn long-term dependencies from sequential data and the ability to remember along the sequence makes it applicable in many computer vision tasks, including semantic segmentation. **Table 2.4** shows RNN-based methods.

Table 2.4: **Recurrent Neural Network based Methods**

Category	Strategy / Structure		Corpus	Original Architecture	Testing Benchmark	Code Available
Recurrent Neural Network	Recurrent Convolution Neural Network ($_R$CNN) Pinheiro and Collobert (2014)		Feed-Forward Approach: Models non-local class dependencies in a scene from the raw image (Extract contextual information).	LeNet	Stanford Background SIFT Flow	-
	Directed Acyclic Graph RNNs	DAG-RNNs Shuai et al. (2016)	Model the contextual dependencies of local features. Class Weighting Function that attends to rare classes.	VGGNet + RNN	SiftFlow, CamVid, Barcelona	-
		DAG-RNNs Shuai et al. (2017)	Model long-range semantic dependencies for graphical structured images. Class Weighting Function that attends to rare classes.	VGGNet + RNN	Sift Flow, Pascal Context COCO Stuff	-
	ReSeg: Recurrent Segmentation Visin et al. (2016)		Modified ReNet Recurrent Layer: Composed by multiple RNNs. Gated Recurrent Unit (GRU) or LSTM	ReNet + RNN	CamVid, Oxford Flower, Weizmann Horse	YES
	Multi-level Contextual Recurrent Neural Networks (MCRNNs) Fan et al. (2018)		CRNNs encode three contextual cues (local, global and GIST). Attention model is adopted to improve effectiveness.	VGGNet + RNN	CamVid, KITTI, SiftFlow, Stanford-background, Cityscapes	-
	Multi-level Graph Convolutional Recurrent Neural Network (MGCRNN) Jiang et al. (2021)		Formulates graph neural network (GNN) as a RNN to reconstruct pairwise relationships between pixels and aggregate multi-level contextual information.	VGGNet GNN	Pascal VOC, Cityscapes	-
	Recurrent model for semantic instance segmentation Salvador et al. (2017)		Encoder/Decoder based Recurrent Neural Network Encoder: Feature extractor Decoder: Convolutional LSTM, predicting one instance at a time	ResNet + Convolutional LSTM	Pascal VOC 2012, Cityscapes, CVPPP Plant Leaf Segmentation	YES

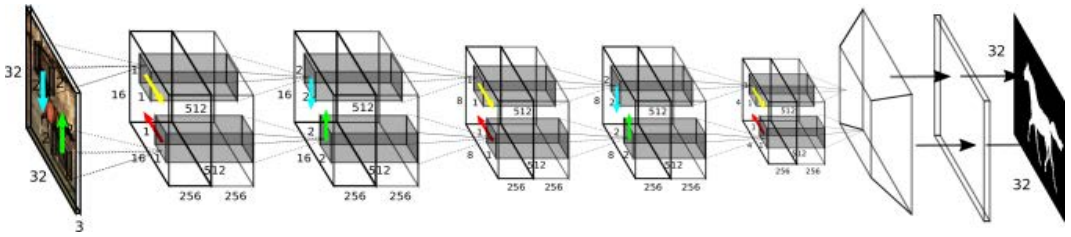


Figure 2.6: ReNet Network Visin et al. (2016)

Pinheiro and Collobert (2014) proposed a convolutional neural network based on a recurrent architecture ($_R$CNN). $_R$CNN is a sequence of shallow networks sharing same weights, each instance of which uses the down-scaled input image and prediction maps of the previous instance of the network and automatically learns to smooth its predicted labels. Fan et al. (2018) presented the contextual RNNs for scene labeling. The network can capture long-range dependencies (GIST, local and global features) in an image. These features are fused (after upsampling) using an attention model Chen et al. (2016b). Salvador et al. (2017) introduce an encoder/decoder based recurrent neural network architecture for semantic instance segmentation. Its architecture is very similar to the FCN Long et al. (2015) architecture (encoder: feature extractor) using skip-connection, except for the decoder part, which is a recurrent network (convolutional LSTM Shi et al. (2015)) that predicts and outputs one instance (object in the image) at a time. Visin et al. (2016) developed an RNN-based architecture for semantic segmentation, codenamed ReSeg, to model the structural information of local generic features extracted from CNNs. The model is a modified and extended version of ReNet Visin et al. (2015). The proposed recurrent layer consists of multiple RNNs Cho et al. (2014)Hochreiter and Schmidhuber (1997) that search the image horizontally and vertically in both directions (hidden state output), encode local features and provide relevant global information. The ReNet layers are stacked on top of the output of an FCN. **Figure 2.6** shows the architecture of the ReNet network. Shuai et al. (2016) use graphical RNNs (Directed Acyclic Graph - Recurrent Neural Network or DAG -RNN) to model long-range contextual dependencies of local features in the image for semantic segmentation. They proposed a new class weighting function to improve the accuracy of detecting non-frequent classes. Later, Shuai et al. (2017) proposed a DAG-RNN network to model long-range semantic dependencies for graph structured images (DAG -structured). Their proposed segmentation network consists of three modules: local region representation (using a pre-trained CNN), context aggregation (using DAG -RNN), and feature map upsampling (deconvolution network). In addition, class-weighted loss was used in training to solve the class imbalance problem or to account for rare classes. Recently, Jiang et al. (2021) introduced a segmentation model called Graph Convolutional Recurrent Neural Network (GCRNN), which formulates a Graph Neural Network (GNN) as an RNN to reconstruct pairwise relationships between pixels and aggregate multi-level contextual information.

A recurrent neural network (RNN) can be very beneficial in semantic segmentation; it has recurrent connections (ability to retain previous information) and the ability to capture context in an image by modeling long-range semantic dependencies for the image.

2.2.1.4/ UPSAMPLING / DECONVOLUTION BASED METHODS

Convolutional neural network models have the ability to automatically learn high-level features via layer-by-layer propagation, while losing spatial information. One deep understanding is that spatial information lost in the down-sampling operation can be recovered by upsampling and deconvolution. Secondly, a reconstruction technique is developed to increase spatial accuracy and a refinement technique is developed to merge low level and high level features. **Table 2.5** shows upsampling / deconvolution based methods.

Noh et al. (2015) used this idea and developed a network model by learning a deconvolution network. The convolutional network reduces the size of the activation's by feed forwarding, and the deconvolution network increases the activation's by combining unpooling and deconvolution operations. Wang et al. (2016) proposed an object-based semantic segmentation (OA-Seg) method using two networks: an object proposal network (OPN) for predicting object bounding boxes and their objectness scores, and a lightweight deconvolution neural network (Light-DCNN) for up-sampling feature maps to higher resolution. Long et al. (2015) introduced the first Fully Convolutional Network (FCN) and achieved a breakthrough in Deep Learning based semantic segmentation. FCN architectures have become the standard in semantic segmentation; most methods use the FCN architecture. FCN covers the classification network Krizhevsky et al. (2012)Szegedy et al. (2015)Mamalet and Garcia (2012) into a fully convolutional network and generates a probability map for an input of arbitrary size. FCN recovers spatial information from down-sampling layers by adding upsampling layers to the standard convolutional network. They defined a skip architecture (shallow fine layer) that combines semantic information from a deep coarse layer with appearance information to produce a precise and deep segmentation. The basic idea was to re-architect and fine-tune the classification model (image classification) to efficiently learn from whole image inputs and whole image ground truths (semantic segmentation prediction). This leads to extending these classification models to segmentation and improving the architecture with combinations of multiple resolution layers. **Figure 2.7** shows the FCN architecture.

Badrinarayanan et al. (2017) present an encoder-decoder structure for deep fully convolutional neural network called SegNet. The encoder network has the same topology as VGG without fully connected layers, followed by a decoder network (from Ranzato et al. (2007)) for pixel-wise classification. SegNet achieves higher resolution than in FCN by using a set of decoders, each corresponding to an encoder. One key feature of Seg-

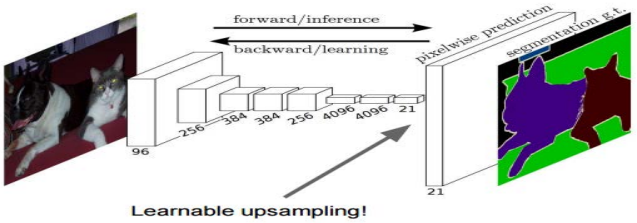


Figure 2.7: FCN: Segmentation Network Long et al. (2015)

Net is that it transfers information directly, rather than convolving it. SegNet has been one of the best models for handling image segmentation problems, especially for scene segmentation tasks. Lin et al. (2017b) proposed a multi-path neural network named refinement network (RefineNet). RefineNet is an encoder-decoder architecture inspired by the residual connection design He et al. (2016) and consists of three components: residual convolution unit (RCU), multi-resolution fusion, and chained residual pooling. The multi-path network uses features at multiple levels, it refines low-resolution features with low-level concentrated features in a recursive manner to produce high-resolution feature maps for semantic segmentation. Vertens et al. (2017) developed an architecture for a semantic motion segmentation network (SMS-Net) consisting of three components: a section that learns motion features from generated optical flow maps, a parallel section that generates features for semantic segmentation, and a fusion section that combines both the motion and semantic features and also learns deep representations for pixel-wise semantic motion segmentation. Islam et al. (2017) presented a refinement structure architecture called Label Refinement Network (LRN). LRN learns the prediction of segmentation labels at multiple levels in the network and gradually refines the results at finer scale. LRN is an encoder-decoder architecture and has monitoring at multiple levels (at each stage of the decoder). Zhao et al. (2018) proposed an image cascade network (ICNet) that efficiently uses low resolution semantic information along with details from high resolution images. The network focuses on fusion of features from multiple layers. They proposed a cascade feature fusion (CFF) which fuses the low feature maps with the high feature maps. Jégou et al. (2017) builds a Fully Convolutional DenseNet FC -DenseNet, extending Huang et al. (2017) by adding an upsampling path and skipping connections to restore full input resolution. Bilinski and Prisacariu (2018) designed an architecture that follows an encoder-decoder strategy. The encoder is based on the ResNeXt architecture and the decoder consists of blocks (dense decoder shortcut connections) that generate semantic feature maps and allow multi-level fusion in a single pass.

Wu et al. (2017) proposed a fully combined convolutional network (FCCN) to improve the upsampling operation of FCN. The network follows a layer-wise upsampling strategy, and after each upsampling operation, the size of the input feature map is doubled. They also proposed a soft cost function to further improve the training. The FCCN was ex-

Table 2.5: **Upsampling / Deconvolution based Methods**

Category	Strategy / Structure			Corpus	Original Architecture	Testing Benchmark	Code Available
Upsampling / Deconvolution	.		Ojectness-Aware Segmentation (OA-Seg) Wang et al. (2016)	Object Proposal Network (OPN) generate object proposals Lightweight deconvolutional neural network (Light-DCNN) for upsampling	VGGNet	PASCAL VOC	-
			Fully Convolutional DenseNet (FC-DenseNet) Jégou et al. (2017)	Built from a down-sampling path, an upsampling path and skip connections. The main goal is to exploit the feature reuses	DenseNet	CamVid Gatech	YES
	Unpooling of Low Level Features or Score Maps	Encoder Decoder	ConvDeconvNet Noh et al. (2015)	Convolution Network: Feature extractor Deconvolution Network: Shape Generartor from the feature extractor	VGGNet	PASCAL VOC	YES
			SegNet Badrinarayanan et al. (2017)	Obtain higher resolution by using a set of decoders one corresponding to each encoder.	VGGNet, DeconvNet	Cityscapes, KITTI, SUN RGB-D, CamVid	YES
			Squeeze-SegNet Nanfack et al. (2018)	DFire Module: Series of concatenation of expand module of SqueezeNet.	SqueezeNet SegNet	CamVid, Cityscapes	-
			Fully Convolutional Network (FCN) Long et al. (2015)	Deep filter consisting (convolution, pooling, activation functions, deconvolution) layers. Upsampling: end-to-end learning by backpropagation from the pixel-wise loss.	Finetuning of AlexNet, VGGNet, GoogLeNet	Cityscapes, CIFAR10, KITTI, PASCAL VOC, CamVid, ADE20K, PASCAL Context, SYNTHIA, Freiburg Forest	YES
			Skip Layer Architecture	Skip (Shallow fine layer) that combines semantic information from a deep, coarse layer with the appearance information to improve segmentation. FCN32s FCN16s FCN8s			
	Reconstruction and Refinement	**Feature Fusion**	Fully Combine Convolutional Network (FCCN) Wu et al. (2017)	Fusing and reusing feature maps Layer by Layer	FCN-VGG	CamVid, PASCAL VOC, ADE20K	-
			Semantic Motion Segmentation Network (SMSNet) Vertens et al. (2017)	Motion feature component: FlowNet2 architectureIlg et al. (2017) Semantic Segmentation component: AdapNet architecture Fusion component: combines both the motion and semantic features	FlowNet, AdapNet	Cityscapes, KITTI	YES
			Dense Decoder Shortcut Connections Bilinski and Prisacariu (2018)	Encoder: ResNeXt architecture A decoder is made up of blocks which generate semantic features maps. Multi-level fusion in single-pass inference	ResNeXt	Pascal VOC, Pascal-Context, Pascal Person-Part, NYUD, CamVid	-
			Image Cascade Network (ICNet) Zhao et al. (2018)	Proposed a cascade feature fusion (CFF) unit	Modified PSPNet	Cityscapes	YES
			Refine Network (RefineNet) Lin et al. (2017b)	Three Components 1. Residual convolution unit (RCU) 2. Multi-resolution fusion 3. Chained residual pooling	ResNet	Cityscapes, ADE20K, NYUDv2, SUN-RGBD, PASCAL VOC & Context	YES
			Patch Proposal Network (PPN) Wu et al. (2020a)	GRNet, consisting 1. Global branch (generates (the preliminary global-level segmentation feature of downsampling) 2. PPN (patch selection) 3. Refinement branch (feature extraction and refinement)	Faster RCNN GRNet	Cityscapes	-
			RGB-D Multi-level Residual Feature Fusion Network (RDFNET) Park et al. (2017)	Multi-modal feature fusion (MMF): the fusion of features (RGB and depth) Multi-level feature refinement: Refining feature	RefineNet	NYUDv2, SUN RGB-D	YES
		Encoder Decoder	Gated Feedback Refinement Network (G-FRNet) Amirul Islam et al. (2017)	Gate Unit: Combines low-resolution features and high-resolution features to produce contextual information. Refinement unit: Generate new label maps with larger spatial dimensions.	VGGNet	CamVid, PASCAL VOC, Horse-Cow Parsing	YES
			Label Refinement Network (LRN) Islam et al. (2017)	Predicts semantic labels at several different resolutions in a coarse-to-fine fashion.	SegNet	CamVid, SUN RGB-D, PASCAL VOC	-

tended by Yang et al. (2019a) to include a highly fused network. The proposed network consists of three main parts: Feature Down-sampling, Combined Feature Upsampling, and Multiple Predictions. The fused network uses the information of the multiple scaled features in the lower layers. Multiple soft cost functions are used to train the proposed model. Inspired by RefineNet, Park et al. (2017) presented an RGB-D fusion network (RDFNet) for semantic segmentation. The proposed architecture consists of two feature fusion blocks: the multi-modal feature fusion (MMF) to fuse features (RGB and depth) in different modalities, and the multi-level feature refinement block to further refine features for semantic segmentation. Amirul Islam et al. (2017) developed Gated Feedback Refinement Network (G-FRNet), an encoder-decoder style architecture. The proposed gated mechanism (Gate Unit) takes two feature maps in sequence, i.e., low-resolution features with larger receptive fields and high-resolution features with smaller receptive fields, and combines them to generate contextual information. The feature maps with different spatial dimensions generated by the encoder network pass through the gate unit before being fed to the decoder (feedback refinement network). The refinement network gradually refines the feature label maps. Nanfack et al. (2018) introduced an encoder-decoder architecture based on Squeeze-SegNet. The encoder module is a SqueezeNet architecture Iandola et al. (2016) (using the Fire module and removing the Average Pooling layer) inspired by SegNet and removing all fully connected layers of the VGG. The Squeeze decoder module is the inversion of the Fire module and the convolutional layers of SqueezeNet. Recently, Wu et al. (2020a) design a Patch Proposal Network (PPN), which is a binary classification network that selects patches that contain object edges or details that need refinement, while patches contain only background or flat regions that are more likely to be ignored. They also embed the PPN in a global-local network that contains a global branch and a refinement branch, called GRNet. GRNet consists of the global branch (generates the preliminary global-level segmentation feature of down-sampling), the PPN (patch selection) and the refinement branch (feature extraction and refinement). The global-level feature and the refined local feature are fused to produce the final segmentation.

2.2.1.5/ Increase Resolution of Feature Based Methods

Another type of method is to restore spatial resolution by using atrous convolution Chen et al. (2014) and dilated convolution Yu and Koltun (2015), which can produce high-resolution feature maps for dense prediction. The dilated convolution accommodates another parameter ”dilation rate” (which describes the space between values in a kernel) in the convolutional layer and has the ability to expand the receptive field without losing resolution. **Table 2.6** shows the increase in resolution of feature-based network models.

Chen et al. (2014) of Google proposed a deep convolutional neural network model called DeepLab. Instead of using deconvolution, they proposed Atrous ("holes") convolution. The Atrous algorithm was originally developed by Holschneider et al. (1990) for computing the undecimated wavelet transform (UWT). The DeepLab architecture is similar to that of Long et al. (2015) with some modifications, converting fully-connected layers to convolutional layers, using a stride of 8 pixels, skipping sub-sampling after the last two pooling layers, and modifying the convolutional filters in the layers (increasing the length of the last three convolutional layers by twice and the first fully connected layer by four times) by introducing zeros. The proposed method is combined with fully connected conditional random fields (CRF) and is able to efficiently generate semantically accurate predictions and detailed segmentation maps. Yu and Koltun (2015) developed a convolutional network module for dense prediction that uses dilated convolutions to combine multi-scale contextual information without losing resolution, and to analyze re-scaled images for semantic segmentation.

This module can be plugged into existing architectures at any resolution. **Figure 2.8** shows an example of a dilation convolution with different dilation rates defining the distance between values in a kernel. Treml et al. (2016) proposed an encoder-decoder structured architecture (SQNet). The encoder is a modified SqueezeNet architecture Iandola et al. (2016), called "Fire", consisting of convolutional and pooling layers. The decoder is based on a parallel dilated convolution layer. Wu et al. (2016b) present a Fully Convolutional Residual Network (FCRN), a new network for generating feature maps of arbitrary higher resolution without changing the weights. They proposed a method to simulate a high-resolution network with a low-resolution network, and an online bootstrapping method for training. Chen et al. (2017a) proposed the Atrous Spatial Pyramid Pooling (ASPP) module, which consists of multiple parallel Atrous convolutional layers with different sampling rates to strongly segment objects at multiple scales. **Figure 2.9** shows an example of ASPP. The proposed network is based on the state-of-the-art ResNet-101 image classification DCNN. They combine the network with a fully connected Conditional Random Field (CRF) to improve object boundary localization.

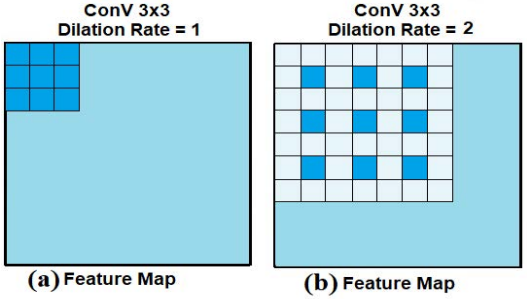


Figure 2.8: Dilated convolution with size of 3×3 with different dilation rates. (a) dilation rate = 1, receptive field = 3×3 (b) dilation rate = 2, receptive field = 7×7.

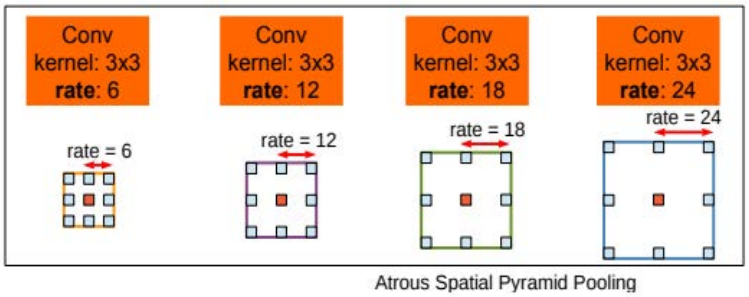


Figure 2.9: Atrous Spatial Pyramid Pooling (ASPP) Chen et al. (2017a)

Yu et al. (2017) introduced another deep neural network called Dilated Residual Network (DRN), a ResNet like architecture where a subset of the inner sub-sampling layers are replaced by dilation Yu and Koltun (2015) to increase resolution. Removing sub-sampling means removing some of the inner layers, which increases the downstream resolution and reduces the receptive field in the downstream layers. They also propose an approach to remove the gridding artifacts introduced by dilation (degridding), which further improves the performance. Later, Chen et al. (2017b) revisited atrous convolution and proposed a new system network called DeepLab V3. They designed new modules in which atrous convolution operates in cascade or parallel (spatial pyramid pooling, as shown in **Figure 2.10 (a)**) to capture the multi-scale context by adopting multiple atrous rates, and used batch normalization for training. The main idea was to duplicate multiple copies of the final block in ResNet and cascade them.

Wang et al. (2018) proposed a method called design dense up- sampling convolution (DUC). The basic idea of DUC is to transform the label map into a smaller label map with multiple channels (dividing the label map into equal sub-parts with the same height and width as the incoming feature map). They also proposed a Hybrid Dilated Convolution

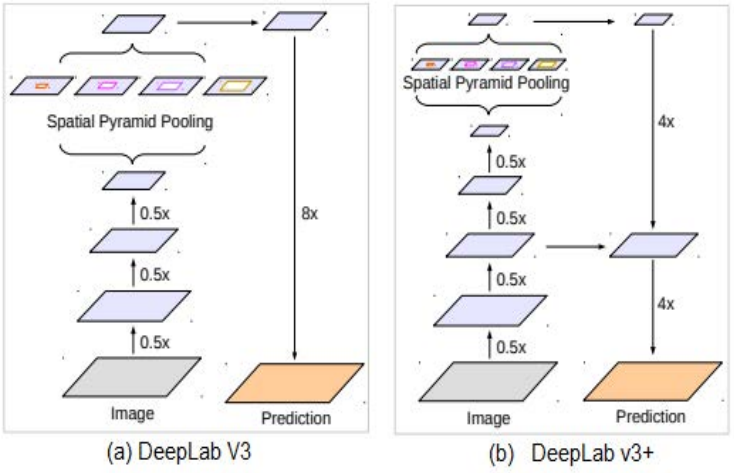


Figure 2.10: DeepLabV3 and DeepLabV3+ Chen et al. (2018)

(HDC) framework in the encoding phase, which effectively enlarges the receptive fields of the network to aggregate global information. Lately, DeepLab V3+ was introduced in Chen et al. (2018), which is the extended version of DeepLab V3. Inspired by Alvarez et al. (2012), the authors proposed a decoder module in which the encoder features are up-sampled by a factor of 4 instead of 16 as in Chen et al. (2017b) and then concatenated with the corresponding low-level features from the network backbone with the same spatial resolution, as shown in **Figure 2.10 (b)**. They adopted the Xception model Chollet (2017) and applied the depth-wise separable convolution (to reduce the computational complexity) to both Atrous Spatial Pyramid Pooling (ASPP) and the decoder modules. A detailed network structure is presented by Gaihua et al. (2021). They introduced a self-attention module based on the serial-parallel structure combined with dilated convolution instead of downsampling. The module improves the receptive field of the network,

Table 2.6: Increase Resolution of Features based Methods

Category	Strategy / Structure		Corpus	Original Architecture	Testing Benchmark	Code Available
.	.	DeepLab Chen et al. (2014)	Atrous ('Holes') Convolution	FCN-VGG	Cityscapes, PASCAL VOC	YES
Increase Resolution of Features	Atrous Convolution	DeepLabV2 Chen et al. (2017a)	Atrous Spatial Pyramid Pooling (ASPP). Method effectively enlarge the field of view of filters to incorporate multi-scale context.	FCN-ResNet	Cityscapes, PASCAL VOC, COCO	YES
		DeepLabV3 Chen et al. (2017b)	Rethink Atrous Convolution Augment the Atrous Spatial Pyramid Pooling (ASPP).	DeepLabV2	Cityscapes, PASCAL VOC	-
		DeepLabV3+ Chen et al. (2018)	Encoder Decoder Approach Xception	DeepLabV3	PASCAL VOC	YES
	Dilated Convolution	. Dilated Convolutions Module Yu and Koltun (2015)	Rectangular Prism convolutional layers, with no pooling or subsampling for multi-scale context aggregation .	VGGNet	Cityscapes, PASCAL VOC	YES
		SQ Network Treml et al. (2016)	Fire module: modified SqueezeNet Parallel dilated convolution layer. Refinement module: SharpMask approach	SqueezeNet	Cityscapes	-
		Hybrid Dilated Convolution (HDC) Wang et al. (2018)	Dense Upsampling Convolution (DUC) by TuSimple.	ResNet + DUC	KITTI, PASCAL VOC	YES
		Series-parallel Structure Self-attention Network Gaihua et al. (2021)	Self-Attention Module: Based on the serial-parallel structure combined with dilated convolution	ResNet	Cityscapes, PASCAL VOC	-
.		Dilated Residual Network (DRN) Yu et al. (2017)	Replacing dilated convolutions layers into ResNet model.	ResNet	Cityscapes	YES
	Fully Convolutional Residual Network (FCRN) Wu et al. (2016b)		Method to simulate a high resolution network with a low resolution network. Enlarge the field-of-view (FoV) of features. Online bootstrapping method for training.	ResNet + FCN DeepLab	Cityscapes, PASCAL VOC	-
	Efficient semantic segmentation with pyramidal fusion (SwiftNet) Oršić andŠegvić (2021)		Multi-scale architecture with pyramidal fusion. Spatial Pyramid Pooling (SPP). Increasing the penalty for boundary pixels.	ResNet MobileNet V2	Cityscapes, ADE20k, CamVid, Mapillary Vistas	YES

which can simulate reliable long-range correlation for similar features, compensate for the missing feature, and improve the recognition accuracy of semantic segmentation.

Compared to regular convolution with larger filters, atrous convolution allows to effectively enlarging the field of view of the filters without increasing the number of parameters or computational complexity. Dilated convolution is a simple but powerful alternative to deconvolution for dense prediction tasks.

2.2.1.6/ Enhancement of Features Based methods

Enhancement of feature based methods include extracting features at multiple scales or from a sequence of nested regions. In deep networks for semantic segmentation, CNNs are applied to square image patches, often referred to as fixed-size kernels, centered on each pixel, where each pixel is labeled by observing a small region around it. The network covering a large and wide context (size of the receptive field) is essential for better performance, which can be achieved but increases the computational complexity. Feature extraction at multiple scales or extraction from a sequence of nested regions can be considered while ensuring computational efficiency. **Table 2.7** shows the enhancement of feature-based network models.

Farabet et al. (2013) proposed a method that extracts multiscale feature vectors from the image pyramid (Laplacian pyramid version of the input image) using the multiscale convolutional network shown in **Figure 2.11**. Each feature vector encodes regions with multiple sizes centered on each pixel location, covering a wide context. Liu et al. (2016) proposed the strategy called multi-scale Patch Aggregation (MPA). The proposed network generates multiscale patches for object parsing, achieves segmentation and classification for each patch at the same time and aggregates them to infer objects. Mostajabi et al. (2015) present a feed forward classification method called Zoom-Out using Superpixels (SLIC Achanta et al. (2012)). It extracts features from different levels (local level: superpixel itself; distant level: regions large enough to cover fractions of the object or the entire object; scene level: entire scene) of the spatial context around the superpixel to contribute to the labeling decision at that superpixel. Then, a feature representation is

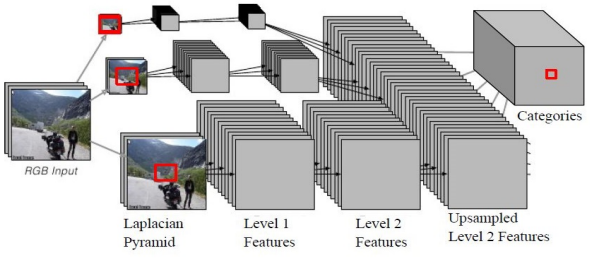


Figure 2.11: Multiscale CNN for scene parsing Farabet et al. (2013)

Table 2.7: **Enhancement of Features based Methods**

Category	Strategy / Structure		Corpus	Original Architecture	Testing Benchmark	Code Available
Enhancement of Features	Multi-scale Features Extraction	Multi-Scale Network Farabet et al. (2013)	Multi-scale Convolutional Network extract dense feature vectors that encode regions of multiple sizes centered on each pixel. Multiple post-processing methods for labeling.	LeNet	Sift Flow, Barcelona, Stanford Background	-
			Learn multi-scale features using the image depth information.	LeNet	NYUDv2	-
		Multi-scale Patch Aggregation (MPA) Liu et al. (2016)	Multi-scale Patch Generator: Cropping corresponding feature grids from Image, and aligning these grids to improve the generalization ability. A strategy is proposed to assign the classification and segmentation labels to the patches.	VGG-16	PASCAL VOC, COCO	-
		DeepLab Attention Model Chen et al. (2016c)	Learns to weight the multi-scale features according to the object scales presented in the image, then for each scale outputs a weight map which weights feature pixel by pixel.	DeepLab	PASCAL VOC, COCO	-
		Pyramid Scene Parsing Network (PSPNet) Zhao et al. (2017)	Pyramid pooling module consists of the large kernel pooling layers for global scene prior construction	ResNet Dilated FCN	ImageNet, Cityscapes, ADE20K, PASCAL VOC	YES
		Cascade Dilated Convolutions Network Vo and Lee (2018)	Cascading dilated convolutions (consecutive layers connection) to extract dense features. Feature fusion through Maxout Layer (Maxout Network Goodfellow et al. (2013))	Dialted-ResNet FCN-VGG	PASCAL VOC	-
		Context Aggregation Network Yang et al. (2021)	Reformulating global aggregation and local distribution (GALD) blocks. Fusion block (FFM) to assists in feature normalization and selection for optimal scene segmentation.	MobileNetV3 Dialted-ResNet	Cityscapes, UAVid	-
		Multiply Spatial Fusion Network (MSFNet) Si et al. (2019)	Multi-features Fusion Module (MFM): Obtain spatial information and enlarge receptive field.	ResNet	Cityscapes, Camvid	-
		Context Contrasted Local (CCL) Model Ding et al. (2018)	CCL: Consists of several chained context-local blocks to make multi-level context contrasted local features. Gate Sum: Fusion strategy to aggregate appropriate score maps.	ResNet	Pascal Context, SUN-RGBD, COCO Stuff	-
	Feature Extraction from sequence of nested regions	Cascaded Feature Network (CFN) Lin et al. (2017a)	Context-aware Receptive Field (CaRF): to aggregate convolutional features of local context into strong features.	FCN + RefineNet	NYUDv2, SUN-RGBD	-
		SEgmentation TRansformer (SETR) Zheng et al. (2021)	Self-attention based encoder: Fully attentive feature representation encoder by sequentializing images. Three different decoder designs; 1. Naive upsampling 2. Progressive UPsampling (PUP) 3. Multi-Level feature Aggregation (MLA).	FCN	Cityscapes, ADE20K, Pascal Context	YES
		Zoom Out Mostajabi et al. (2015)	Zoom out features construction using superpixels (SLIC Method) from different levels of spatial context Local Level: Superpixel itself Distant Level: Regions large enough to cover fractions of an object or entire object. Scene Level: Entire scene Combining features across levels rather than predicting.	VGG-16	PASCAL VOC	-

computed at each level and all features are combined before being fed to a classifier. The authors Yang et al. (2021) propose Context Aggregation Network, in which they design a high-resolution branch for effective spatial detail and a context branch with lightweight

versions of global aggregation and local distribution (GALD) blocks strong enough to capture both long-range and local contextual dependencies required for accurate semantic segmentation. However, the proposed module is computationally expensive and requires significant GPU memory for execution.

Chen et al. (2016c) proposed an attention-based model with the ability to choose which part of the input to look at each time to accomplish the task. The proposed attention model learns to weight the multiscale features according to the object scales in the image (e.g., the model learns to put large weights on features in a coarse scale for large objects). Then, for each scale, the attention model outputs a weight map that weights the features pixel by pixel, and the weighted sum of the weight maps generated by the FCN across all scales is then used for classification. Zheng et al. (2021) presented a SEgmentation TRansformer (SETR), an alternative perspective by treating semantic segmentation as a sequence-to-sequence prediction task. Their idea is to encode an image as a sequence of patches using a design transformer (inspired by natural language processing (NLP) Devlin et al. (2018)) that models the global context in each layer. They present three different decoder designs with different complexity. Zhao et al. (2017) present a Pyramid Scene Parsing Network (PSPNet) for semantic segmentation that enables multi-scale feature ensembling. They introduced the pyramid pooling module, which consists of large kernel pooling layers shown in **Figure 2.12**. This module empirically proves to be an effective global contextual prior that contains information with different pyramid scales and varies between different sub-regions. It concatenates the feature maps with the upsampled output of the parallel pooling layers. This idea is also known as intermediate supervision. The representations are fed into a convolutional layer to get the final per-pixel prediction.

Vo and Lee (2018) developed a deep network architecture with multi-scale dilated convolution layers to extract multi-scale features from multi-resolution input images. The basic idea is to cascade dilated convolutions (connecting successive layers), where each layer achieves a denser feature map at a higher rate than the previous one. All feature maps are then brought to the same resolution and fused into a maxout layer to obtain the most

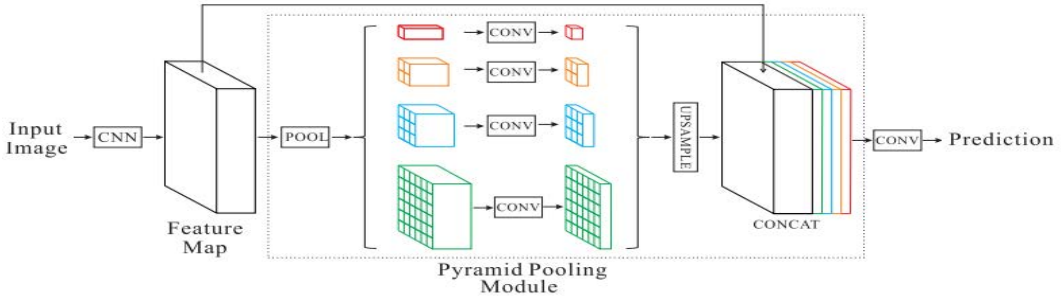


Figure 2.12: Pyramid Scene Parsing Network (PSPNet) Zhao et al. (2017)

driven and leading features from all feature maps. Lin et al. (2017a) proposed a network called cascaded feature network (CFN). It uses depth information and divides the image into layers representing visual characteristic of objects and scenes (multi-scene resolutions). The proposed contextual receptive field CaRF (superpixel based) aggregates convolutional features of the local context into strong features. The CaRF generates contextual representations, large superpixels for low scene resolution regions and finer superpixels for higher scene resolution regions. Ding et al. (2018) presented a context-contrasted local (CCL) model to obtain multiscale features (both contextual and local). Instead of using a simple sum, they proposed a Gate-Sum fusion strategy to aggregate appropriate score maps, which allows a network to choose a better and desired scale of features. Lately, Si et al. (2019) introduced a multi-features Fusion Module in their proposed model Multiply Spatial Fusion Network (MSFNet), which uses Class Boundary Supervision to process the relevant boundary information. The module lets all feature maps of different scales merge with larger ones to increase the receptive field and gain more spatial information

Several methods aimed to capture features with multiple scales, with features at higher layers containing more semantic meaning and less location information. Combining the advantages of multi-resolution images and multi-scale feature descriptors to extract both global and local information in an image without losing resolution improves the performance of the network.

2.2.1.7/ SEMI AND WEAKLY SUPERVISED CONCEPT

CNNs become deeper by increasing the depth and breadth (the number of levels of the network and the number of entities at each level). Deep CNNs require a large dataset and massive computational power for training. Manual collection of labeled datasets is time consuming and requires huge human effort. To reduce this effort, semi-supervised or weakly supervised methods are applied using deep learning techniques. **Table 2.8** shows semi and weakly supervised network models used for semantic segmentation.

The work of Pathak et al. (2014) is the first to address the fine-tuning of CNNs pre-trained for object recognition using image-level labels in a weakly supervised segmentation context. They presented a fully convolutional network method based on a Multiple Instance Learning (MIL -FCN) Maron and Lozano-Pérez (1998), i.e., learning a pixel-level semantic segmentation from weak image-level labels indicating the presence or absence of an object. They proposed a pixel-level multi-class loss inspired by the binary MIL scenario. Pinheiro and Collobert (2015) proposed a weakly supervised approach to generate pixel-level labels from image-level labels using the Log-Sum-Exp (LSE) Boyd and Vandenberghe (2004) method, which assigns the same weight to all pixels of the image during

training. Papandreou et al. (2015) presented a weakly and semi-supervised learning method that uses weak annotations, either alone or in combination with a small number of strong annotations. They developed a method called Expectation Maximization (EM) for training DCNN from weakly annotated data.

Hong et al. (2015) introduced a semi-supervised method (DecoupledNet) that uses two separate networks, one for classification (classifies the object label) and the other for segmentation (to obtain a figure-ground segmentation for each classified label). Dai et al. (2015) developed a method based on bounding box annotations (BoxSup). The unsupervised region proposal method (selective search Uijlings et al. (2013)) is used to generate segmentation masks, and these masks are used to train the convolutional network. The proposed BoxSup model trained with a large set of boxes increases the object recognition accuracy (the accuracy at the centre of an object) and improves the object boundaries. Oh et al. (2021) introduced a new pooling method for weakly-supervised semantic segmentation (WSSS) using bounding box annotations that allows to generate high-quality pseudo-ground truth labels. Luo et al. (2017) presented a weakly and semi-supervised dual image segmentation (DIS) learning strategy that performs segmentation (capturing the accurate object classes) and reconstruction (accurate object shapes and boundaries). The idea is to predict tags, label maps from an input image and reconstruct images using the predicted label maps. Saleh et al. (2016) proposed a weakly supervised segmentation network with built-in foreground/background prior. The main idea is to extract localization information directly from the network itself (foreground/background mask extraction). Later, Saleh et al. (2018) extended their work to obtain multi-class (class-specific) masks by fusing foreground/background masks with information extracted from a weakly supervised localization network inspired by Zhou et al. (2016a). Saito et al. (2017) present a method that uses feature maps extracted from a pre-trained dilated ResNet with built-in priors for semantic segmentation. They proposed a superpixel clustering method to generate road clusters (to select the largest cluster in the bottom half of the image), which are used as labels to train the CNN for segmentation. Barnes et al. (2017) developed a weakly supervised method for autonomous driving applications to generate a large set of labeled images (from multiple sensors and data collected during driving) containing path proposals without manual annotation. Ye et al. (2018) proposed a method for learning convolutional neural network models from images with three different types of annotations, i.e., image-level labels for classification, box-level labels for object detection and pixel-level labels for semantic segmentation. They proposed an annotation-specific loss module (with three branches, each branch with a different loss function) to train the network for each of the three different annotations. Xu et al. (2021) proposed an atrous convolutional feature network that contains two important modules, namely an atrous convolution cascade module (to obtain more spatial details) and an atrous convolution pyramid module (to capture multi-scale contextual information).

Table 2.8: **Semi and Weakly Supervised based Methods**

Category	Strategy / Structure		Corpus	Original Architecture	Testing Benchmark	Code Available
Weakly and Semi Supervised	Image Level Labels	Multiple Instance Learning (MIL-FCN) Pathak et al. (2014)	Multi-class pixel-level loss inspired by the binary MIL scenario.	VGG	PASCAL VOC	-
		Aggreg-LSE Pinheiro and Collobert (2015)	An approach to produce pixel-level labels from image-level labels using Log-Sum-Exp (LSE) Boyd and Vandenberghe (2004).	VGG	PASCAL VOC	-
		Utilization of Heterogeneous Annotations: DecoupledNet Hong et al. (2015)	Classification Network: Identifies labels Segmentation Network: Produces pixel-wise figure-ground segmentation corresponding to each identified label. Bridging layers connecting the two Networks (Decoupling).	VGG	PASCAL VOC	YES
		Utilization of Heterogeneous Annotations: WSSL Papandreou et al. (2015)	Expectation Maximum (EM) Module for fast training under both weakly and semi-supervised settings.	DeepLab	Cityscapes, PASCAL VOC	YES
		Simple to Complex (STC) Huang et al. (2018c)	A progressively training strategy is proposed by incorporating simple-to-complex images with image-level labels.	VGG + DeepLab	PASCAL VOC	-
		Dual Image Segmentation DIS Luo et al. (2017)	Segmentation: Predict tags and label maps from the image (captured the accurate object classes). trtruction: The reconstruction of images using predicted label maps (accurate object shapes and boundaries).	ResNet	PASCAL VOC	-
		Adversarial Learning: SW-GAN Souly et al. (2017)	Generative Adversarial Network framework which extends the typical GAN to a pixel-level prediction.	VGG	PASCAL VOC, SiftFlow, StanfordBG, CamVid	-
		Adversarial Learning: Self-Attention Generative Learning Zhang et al. (2020)	Self-attention mechanism for GAN. Spectral normalization to stabilize the training of the discriminator.	DeeplabV2 ResNet	PASCAL VOC Cityscapes	-
		Adversarial Learning: Semi-Adv Hung et al. (2018)	Propose a fully convolutional discriminator that learns to differentiate between ground truth label maps and probability maps of segmentation predictions.	DeeplabV2	PASCAL VOC, Cityscapes	YES
		Segmenting Path Proposals Barnes et al. (2017)	Weakly-supervised approach to segmenting proposed paths for a road vehicle Method for generating a large amount of labeled images without any manual annotation.	SegNet	KITTI, Oxford	-
		Built-in Feature Extraction Approach: Fg/Bg Masks Saleh et al. (2016)	Weakly-supervised segmentation network with built-in Foreground/Background Prior "Information extracted from a pre-trained network".	VGG-16	PASCAL VOC	-
		Built-in Feature Extraction Approach: Multi-Class Mask Saleh et al. (2018)	Foreground/background mask combined to generate the class-specific mask Multi-Class Prior.	VGG-16	PASCAL VOC	-
		Built-in Feature Extraction Approach: Superpixel Clustering Method Saito et al. (2017)	Pre-trained Dilated ResNet for Feature extraction SuperPixel Align Method (FH Superpixel) Road Feature Clustering (K-Means).	DRN + SegNet	Cityscapes	-
		Multi-Dilated Convolutional (MDC) Wei et al. (2018)	Multi-Dilated Convolutional (MDC) Blocks: Produce dense object localization maps which can be utilized for segmentation both in weakly and semi-supervised manner.	VGG + DeepLab	PASCAL VOC	-
		Atrous Convolutional Feature Network (ACFN) Xu et al. (2021)	Atrous Convolution Cascade (ACC) and Atrous Convolution Pyramid (ACP) modules: Produce dense object localization maps, utilized for segmentation.	VGG + DeepLab	PASCAL VOC COCO	-
	Multi-Level Labels	Diverse Supervision: Annotation-Specific FCN Ye et al. (2018)	Annotation-Specific Loss Module Image-level labels for classification Box-level labels for object detection Pixel-level labels for semantic segmentation	FCN	PASCAL VOC	-
	Bounding Box	Boxsup Dai et al. (2015)	The semi-supervised approach based on bounding box annotations Uses SelectiveSearch : to generate segmentation masks. Iterate between an automatically generating region proposals and training convolutional network	FCN	PASCAL VOC, CONTEXT, MS COCO	-
		WSSS Oh et al. (2021)	Dubbed Background-Aware Pooling (BAP): Focuses more on aggregating foreground features inside the bounding boxes using attention maps.	FCN	PASCAL VOC, CONTEXT, MS COCO	-

Souly et al. (2017) developed a semi-supervised semantic segmentation method using adversarial learning inspired by Generative Adversarial Networks (GANs) Goodfellow et al. (2014a). Later, Emre Yurdakul and Yemez (2017) proposed a similar approach consisting of two sub-networks; the segmentation network (for generating class probability maps) and the discriminator network (for generating spatial probability maps with both labeled and unlabeled data). A mechanism for self-attention is introduced by the Zhang et al. (2020), the network is based on adversarial learning and effectively considers relationships between distant spatial regions of the input image with supervision based on pixel-level ground truth data. Wei et al. (2018) presented a weakly and semi-supervised approach using multiple dilated convolutions. They proposed an augmented classification network with multiple dilated convolutional (MDC) blocks that produce dense object localization maps used for semantic segmentation in both weakly and semi-supervised ways. Huang et al. (2018c) proposed a weakly supervised network that generates labels using the contextual information within an image. They proposed a seeded region growing module to find small and tiny discriminative regions of the object of interest by using image labels to generate complete and precise pixel-level labels that are used to train the semantic segmentation network.

Semi-supervised and weakly supervised learning aim to reduce the effort required for full annotation. These methods improve learning performance using weak annotations in the form of image-level labels (information about which object classes are present) and bounding boxes (coarse object locations).

2.2.1.8/ Spatio-Temporal Based Methods

This subsection will study the deep convolutional networks that use spatial information and temporal information for semantic segmentation. In a video, frames are associated with each other and have temporal information (i.e., features of continuous sequences of frames) that can be useful for semantic interpretation of a video. Spatio-temporal structured prediction can prove useful in both supervised and semi-supervised ways. **Table 2.9** shows Spatio-Temporal based network models for semantic segmentation.

Several methods are proposed in the combination of Recurrent Neural Networks (RNN) and Convolutional Neural Network (CNN) for video segmentation. Fayyaz et al. (2016) presented a full convolutional network Spatio-Temporal Fully Convolutional Network (STFCN) employing spatial and temporal features. They proposed a spatio-temporal module that takes advantage of LSTM to define temporal features. The spatial feature maps of the region in a single image fed into the LSTM establish a relationship with the spatial features of equivalent regions in the images before it. Furthermore, the spatial and temporal information is fed into an dilated convolution network (Yu and Koltun (2015)

Table 2.9: Spatio-Temporal based Methods

Category	Strategy / Structure	Corpus	Original Architecture	Testing Benchmark	Code Available
.	Clockwork FCN Shelhamer et al. (2016)	Clockworks: clock signals that control the learning of different layers with different rates	FCN Clockwork RN	Youtube-Objects, NYUD, Cityscapes	YES
Spatio-Temporal	Auto-Path Aggregation (APANet) Hu et al. (2021a)	APANet: predicting multi-level pyramid features that selectively and adaptively aggregate the task-specific hierarchical spatio-temporal contextual information obtained on the features of each individual level.	Mask R-CNN FPN	Camvid NYUDv2	YES
	Spatio-Temporal FCN Fayyaz et al. (2016)	Spatial-Temporal Module embedding into FCN LSTM to define relationships between image frames	FCN	Camvid NYUDv2	YES
	Spatio-Temporal Data-Driven Pooling (STD2P) He et al. (2017b)	Incorporate superpixels and multi-view information into convolutional networks	FCN	NYUDv2 SUN 3D	-
	Feature Space Optimization (FSO) Kundu et al. (2016)	Optimize the mapping of pixels to a Euclidean feature space used by DenseCRF for spatio-temporal regularization	VGG Dilation	CityScapes, Camvid	YES
	Deep Spatio-Temporal FCN (DST-FCN) Qiu et al. (2018)	Learn spatial-temporal dependencies through 2D FCN on pixels and 3D FCN on voxels	VGG C3D	A2D, CamVid	-
	Gated Recurrent FCN Siam et al. (2017)	Implementation of three gated recurrent architectures RFC-LeNet: Conventional Recurrent Units. RFC-VGG and RFC-Dilated: Convolutional Recurrent Units.	FCN	SegTrack V2, Davis, Cityscapes, SYNTHIA	-
	WSBFSaleh et al. (2017)	Weakly-Supervised Two-stream Network. One stream takes image, and other optical flow to extract the features. RFC-VGG and RFC-Dilated: Convolutional Recurrent Units.	VGG	Cityscapes, CamVid, YouTube-Objects	-
	S3-Net Cheng et al. (2021c)	Locates and segments target sub-scenes, extracts structured time-series semantic features as inputs to an LSTM-based spatio-temporal mode Transformer.	ResNet LSTM	CityScapes, UCF11 HMDB51 MOMENTS	-
	Gated Recurrent Flow Propagation (GRFP) Nilsson and Sminchisescu (2016)	Spatio-Temporal Transformer Gated Recurrent Unit (STGRU) Combining spatial transformer with convolutional-gated architecture.	Dilation LRR	CityScapes, Camvid	-

with minor modifications) for upsampling and fused for semantic predictions (summation operation). He et al. (2017b) proposed the Spatio-temporal data-driven pooling model (STD2P), a method for integrating multi-view information using superpixels and optical flow. The goal of semantic segmentation from multiple views is to exploit the potentially richer information from multiple views with better segmentation than from a single view. Qiu et al. (2018) introduced an architectural model based on 2D/3D FCNs called Deep Spatio-Temporal Fully Convolutional Networks (DST-FCN), which exploits the spatial and temporal dependencies between pixels and voxels. The proposed architecture is a network with two streams, a sequential frame stream (2DFCN for spatial and ConvLSTM for temporal information) and a clip stream (3DFCN based on C3D Tran et al. (2015) developed at voxel level). The authors Cheng et al. (2021c) propose a single-shot segmenta-

tion strategy named S3-Net that locates and segments the target scene into sub-scenes (optimized object regions without background) instead of segmenting all pixels or each candidate object in a frame. The proposed model is an LSTM-based spatio-temporal model based on the structured semantic time series features extracted from the previous segmentation model for activity detection in the video stream.

Some architectures are based on Gated Recurrent Architectures, to overcome the gradient problem. Siam et al. (2017) presented a fully convolutional network based on a gated-recurrent architecture (RFCN). Three different architectures were used following two approaches, conventional recurrent units (RFCLeNet) and convolutional recurrent units (RFC VGG, RFC Dilated), which learn spatio-temporal features with a smaller number of parameters. Nilsson and Sminchisescu (2016) proposed Gated Recurrent Flow Propagation network. They proposed Spatio Temporal Transformer Gated Recurrent Unit (STGRU), which combines the strength of spatial transformer (for optical flow warping) with convolution gated architecture (for adaptive propagation and fusion of estimates). Shelhamer et al. (2016) proposed a network called Clockworks, which is a combination of FCN and clockwork recurrent network Koutnik et al. (2014), where the layers of the network are grouped into stages with different clock rates (either fixed clock rate or adaptive clock) and then fused via skip connections. Saleh et al. (2017) introduced a weakly supervised framework for semantic segmentation of videos that treats both foreground and background classes equally. The basic idea is to treat multiple foreground objects and multiple background objects equally. They propose an approach to extract class-specific heat maps from the classifier that locates the different classes for both without pixel-level or bounding-box annotations. Kundu et al. (2016) proposed a model to optimize the feature space used by the fully connected conditional random field for Spatio-temporal regularization. Recently, Hu et al. (2021a) proposed an adaptive aggregation approach called Auto-Path Aggregation Network (APANet), in which the spatio-temporal contextual information contained in the features of each layer is selectively aggregated using the developed ”auto-path”. The ”auto-path” links each pair of features extracted at different pyramid levels for task-specific hierarchical aggregation of contextual information, which enables selective and adaptive aggregation of pyramid features in accordance with different frames. The APANet can be further optimized together with the mask R-CNN head as a feature decoder and a Feature Pyramid Network (FPN) feature encoder, forming a joint learning system for future instance segmentation predictions.

2.2.1.9/ TRANSFORMER BASED METHODS

The Transformer Vaswani et al. (2017) is encoder decoder structured network that uses multi-head attention mechanisms (MHAM) and point-wise feed-forward (PFF) networks to eliminate recurrence and convolutions, illustrated in Figure 2.13. A stack of six identical

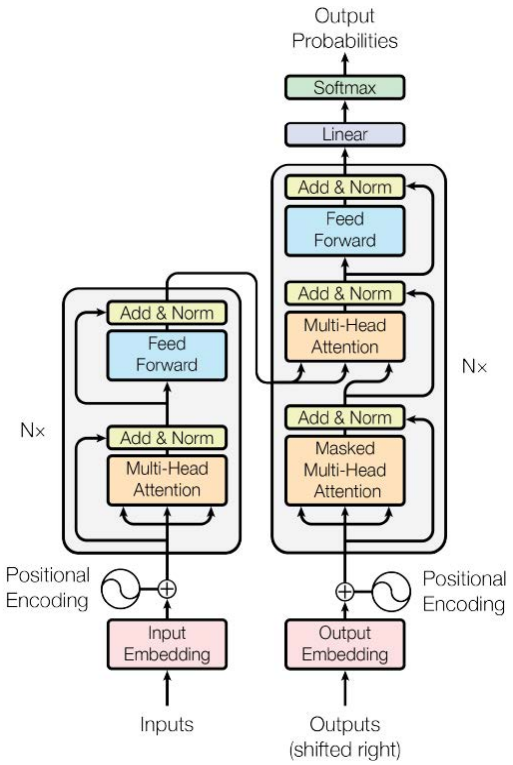


Figure 2.13: The Transformer - model structure. Vaswani et al. (2017)

layers makes up the encoder. Sub-layers are found in every layer. The first is a MHAM, and the second is a simple PFF network. Using a residual connection between each of the two sub-layers and then normalizing the layers. Thus, the output of each sub-layer is a combination of the layer norm and the sublayer's own function. The decoder has six identical layers, just like the encoder. Additionally, the decoder adds a third sub-layer to each encoder layer, which is used to perform multi-head attention on the encoder stack output. Using residual connections around each sub-layer, followed by layer normalization. Further, the decoder stack's self-attention sublayer is tweaked to prevent positions from paying attention to succeeding positions. Because of this masking and the one-position offset of the output embeddings, predictions for position i can only be based on data from positions less than i.

Deep learning models based on transformers have steadily gained prominence in the field of natural language processing (NLP). There have been a number of recent works that have taken these ideas and applied them to computer vision tasks, and achieved good outcomes. Using image patches as input, Dosovitskiy et al. (2020) propose a pure Transformer that achieves SOTA on numerous image classification benchmarks. Other computer vision tasks, such as detection, segmentation, tracking, image generation, and enhancement, have also been well-served by visual Transformers (**ViT**). In the following section, we will look at original visual Transformers and those that are available for the task of segmentation only. These are decomposed into Transformers with patch embed-

Table 2.10: Transformer based Methods

Category	Strategy / Structure		Corpus	Original Architecture	Testing Benchmark	Code Available
Transformer	Patch-Encoding	SETR Zheng et al. (2021)	Progressive upsampling Multilevel feature Aggregation (MLA)	ViT	Cityscapes ADE20K, Pascal Context	Yes
		TransUNet Chen et al. (2021a)	Hybrid CNN-Transformer	ViT U-Net	Synapse multi-organ CT	Yes
		SegFormer Xie et al. (2021)	Positional-encoding-free hierarchical Transformer. Lightweight All-MLP decoder	ViT	Cityscapes ADE20K, COCO Stuff	Yes
	Mask Encoding	Segmenter Strudel et al. (2021)	Point-wise linear mapping. Mask transformer	ViT DETR	Cityscapes ADE20K, Pascal Context	Yes
		MaskFormer Cheng et al. (2021a)	Mask Classification	ResNet DETR	ADE20K, COCO Stuff, Mapillary Vistas, Cityscapes	Yes
		ISTR Hu et al. (2021b)	Low-dimensional Mask embedding. Recurrent Refinement strategy.	R-101-FPN	COCO	Yes
	Object Encoding	Panoptic DETR Carion et al. (2020)	Predictions via Bipartite matching. Non-autoregressive parallel decoding.	ViT FPN	COCO panoptic	Yes
		VisTR Wang et al. (2021b)	Similarity Learning	ResNet-50 DETR	YouTube-VIS	Yes

ding, object embedding, and mask embedding.

Semantic segmentation is formulated in the authors' Strudel et al. (2021) words as a problem of sequencing from one sequence to another. They propose the **Segmenter** transformer architecture, which uses contextual information at every stage of the model. ViT encoder is used to extract image features from the image after it has been divided into patches. The model then treats linear patch embeddings as input tokens through the ViT encoder. Later, the contextualized sequence of tokens is decoded using a point-wise linear mapping of patch embeddings to classification space, which results in the generation of class masks. Wang et al. (2021b) proposed **VisTR**, an end-to-end parallel sequence decoding/prediction framework based on Transformers for video instance segmentation. VisTR uses a bipartite matching loss based on instance sequence level to maintain output order, forcing one-to-one predictions. An encoder-decoder Transformer with 3D position encoding is used to model the similarity of pixel-level and instance-level features. VisTR approaches VIS from a new similarity learning angle. Instance segmentation learns pixel-level similarity while instance tracking learns inter-instance similarity. Based on ViT, Zheng et al. (2021) presented the SEgmentation TRansformer (**SETR**), an extension of the visual Transformer to semantic segmentation tasks. Only the class token is missing from the input-output structure of ViT's transformer encoder, which is based on CNN. More than that, it makes use of multiple decoder styles to accurately classify pixels based on progressive upsampling and multilevel feature aggregation (MLA) decoder styles. SETR shows that the Transformer encoder is a viable option for segmentation, but it requires expensive GPU clusters and additional RAMs due to the number of stack layers and quadratic computational costs associated with the task. Cheng et al. (2021a) developed **MaskFormer**, a parallel Transformer-CNN decoder that uses the set prediction mechanism proposed in DETR to separate mask embeddings and per-pixel features.

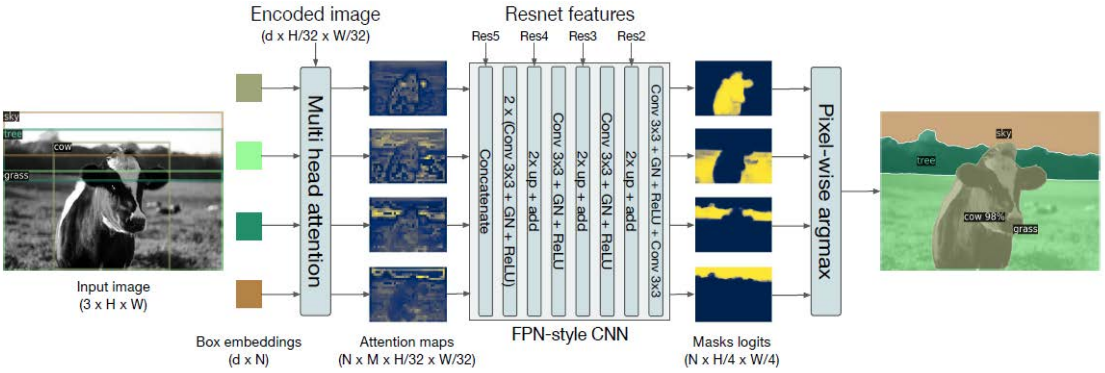


Figure 2.14: An overview of mask head in panoptic DETR Carion et al. (2020).

The model then uses a dot product of the per-pixel embedding from an underlying fully-convolutional network to predict a set of overlapping binary masks. A matrix multiplication is used at the time of semantic inference to combine them and produce the final prediction. Using low-dimensional embeddings instead of raw masks, Hu et al. (2021b) proposed **ISTR** to achieve end-to-end instance segmentation, allowing the training to be completed with a small number of matched samples. In addition, a recurrent refinement strategy is designed that processes detection and segmentation simultaneously by regressing with the embeddings.

DEtection with TRansformer (**DETR**), developed by Carion et al. (2020), is an object detection method that has been applied for panoptic segmentation. It uses an encoder-decoder Transformer as the neck and a Feature Pyramid Network (FPN) style CNN as the prediction head. The model learns a set of object queries which are (similar to the encoder) learned positional encodings, that are appended to zero inputs before being fed in parallel to the Transformer decoder. A self-attention block in the decoder deals with the relationship between decoder embeddings, while a cross-attention block aggregates global features into embeddings. **Figure 2.14** shows an overview of mask head in panoptic DETR. The model performs well on the COCO panoptic benchmark. A **SegFormer** transformer model is presented by Xie et al. (2021), which consists of a hierarchical pyramid Transformer as an encoder that outputs multiscale features (without position encoding) and a lightweight decoder with multiple MLP layers that combines local and global attention to produce the segmentation mask. Chen et al. (2021a) proposed **TransUNet**, the first visual Transformer for medical image segmentation. The structure was designed as a combination of U-Net [128] and Transformer to improve finer details by restoring localized spatial information. It encodes the tokenized image patches before directly upsampling the hidden feature representations to produce a dense output. Because of the low efficiency, SegFormer, DETR, and TransUNet Transformer-based methods cannot be used in real-time applications.

2.2.1.10/ METHODS REFINING PIXEL PREDICTIONS

METHODS USING CRF / MRF

Semantic segmentation involves pixel-by-pixel classification, and such pixel-by-pixel classification often produces unsatisfactory results (poor, incorrect, and noisy predictions) that are inconsistent with the actual visual features of the image Arnab et al. (2018). Markov Random Field (MRF) and its variant Conditional Random Fields are classical frameworks widely used to overcome these problems. They express both unary terms (per-pixel label assignment confidence) and pairwise terms (constraints between adjacent pixels). CNNs can be trained to model unary and pairwise terms to capture contextual information. Context provides important information for scene understanding tasks, such as spatial context, which provides the semantic compatibility/incompatibility relationship between objects, scenes, and situations. CRFs can be a post-processing or end-to-end to smooth and refine pixel prediction in semantic segmentation. They combine class scores from classifiers with the information captured by the local interactions of pixels and edges or superpixels. **Table 2.11** shows network models with CRF.

Krähenbühl and Koltun (2011) proposed a fully connected CRF (DenseCRF) model in which the pairwise edge potentials are defined by a linear combination of Gaussian kernels. The method is based on the mean-field approximation, and message passing is performed using Gaussian filtering techniques Adams et al. (2010a). Methods Noh et al. (2015); Chen et al. (2014); Papandreou et al. (2015); Dai et al. (2015); Saleh et al. (2016); Khoreva et al. (2017); Wei et al. (2018); Saleh et al. (2017) coupled fully connected CRF with their proposed DCNNs to produce accurate predictions and detailed segmentation maps to improve performance. Zheng et al. (2015) formulate a mean-field inference algorithm for dense CRF with Gaussian filtering technique as a recurrent neural network (CRF-RNN) that performs CRF-based probabilistic graphical modeling for structured predictions. **Figure 2.15** shows CRF as an RNN.

Vemulapalli et al. (2016) proposed a model called Gaussian Mean Field (GMF) network that models unary potentials, pairwise potentials and Gaussian CRF inference for the task of semantic segmentation. In the proposed network, the output of each layer is closer to the maximum a posteriori probability (MAP) estimated for the input. Chandra and Kokkinos (2016) presented a Gaussian Conditional Random Field (G-CRF) module using a quadratic energy function that captures unary and pairwise interactions. Lin et al. (2016) introduced a model Context CNN CRF that learns CNNs and CRFs jointly. They formulate a CRF with a pairwise CNN potential to capture the contextual relationship between neighboring patches, and a sliding pyramid pooling (multiscale image network input) to capture the patch background context, which can be combined to improve segmentation. Instead of learning the potentials, Lin et al. (2015) proposed a method that learns CNN

Table 2.11: **Methods using CRF/MRF**

Category	Strategy / Structure			Corpus	Original Architecture	Testing Benchmark	Code Available
CRFs / MRFs			Fully Connected-CRF (DenseCRF) Krähenbühl and Koltun (2011)	Based on mean field approximation, message passing performed using Gaussian filtering techniques.	ResNet	PASCAL VOC	Yes
			CRF-RNN Zheng et al. (2015)	Multiple Mean-field Iterations. Interpretation of dense CRFs as Recurrent Neural Networks (CRF-RNN) combined with CNN.	FCN	PASCAL VOC Cityscapes	-
	Gaussian Conditional Random Field (GCRF)		Gaussian Mean Field (GMF) Network Vemulapalli et al. (2016)	GMF Network: Performing Gaussian mean field inference.	DeepLab	PASCAL VOC ImageNet	Yes
			Quadratic Optimization (QO) Chandra and Kokkinos (2016)	Quadratic Optimization (QO) module	FCN	PASCAL VOC	-
			Convolutional-CRF (ConvCRF) Teichmann and Cipolla (2018)	Inference in terms of convolutions.	ResNet	PASCAL VOC	Yes
	Incorporating Higher Order potentials		Higher-order CRF Arnab et al. (2016)	Object-detection based potentials: Provide Semantic cues for segmentation. Superpixel-based potentials: Encourage label consistency over regions.	CRF-RNN	PASCAL VOC, Context	-
			Structured Patch Prediction (SegModel) Shen et al. (2017)	Integrate segmentation specified features, high order context and boundary guidance.	FCN	PASCAL VOC Cityscapes ADE20K	-
			Deep Parsing Network (DPN) Liu et al. (2015b)	Models Unary term and Pairwise terms in single CNN.	VGG	PASCAL VOC	-
			Deep Parsing Network (DPN) Liu et al. (2015b)	Models Unary term and Pairwise terms in single CNN.	VGG	PASCAL VOC	-
	.		Learning Messages Lin et al. (2015)	CNN message estimators for the message passing inference.	VGG-16	PASCAL VOC	-
	Adelaide	Bounding -box Detection	Adelaide Very Deep FCN Wu et al. (2016a)	Hough transform based approach Online bootstrapping method for training.	FCRN	PASCAL VOC	-
			Context CNN CRF Lin et al. (2016)	Patch-patch context: Formulate CRFs to capture contextual relationship between neighboring patches Patch-background context: Sliding Pyramid Pooling.	VGG-16	PASCAL VOC NYUDv2 Pascal Context Siftflow	-
	incorporate the depth information		Depth-sensitive fully-connected Conditional Random Field (DFCN-DCRF) Jiang et al. (2017) .	Fully-connected CRFs with RGB information and depth information.	FCN	SUN-RGBD	-

message estimators for message passing inference for structured Conditional Random Field (CRFs) predictions. Teichmann and Cipolla (2018) developed a convolutional CRFs method (ConvCRFs) that reformulates message passing inference in terms of convolutions.

Some methods used higher-order potentials (based on object detection or superpixels) modeled as CNN layers when they used mean-field inference and effectively improved

semantic segmentation performance. Arnab et al. (2016) proposed a method in which CRF models unary and pairwise potentials together with higher-order object detector potentials (to provide semantic cues for segmentation) and superpixels (with label consistency across regions) in an end-to-end trainable CNN. Shen et al. (2017) presented a joint FCN and CRF model (SegModel) that integrates segmentation-specific features representing higher-order context and boundary guidance (bilateral-filtering based CRF) for semantic segmentation. Liu et al. (2015b) developed Deep Parsing Network (DPN), which models unary terms and pairwise terms (i.e., higher-order relations and a mixture of label contexts) in a single CNN that achieves high performance by extending the VGG network and adding some layers to model pairwise terms. Jiang et al. (2017) utilize the depth information as complementary information in conditional random fields. They proposed a depth-sensitive fully connected conditional random field combined with a fully convolutional network (DFCN-DCRF). The basic idea is to integrate depth information in Dilated-FCN and Fully Connected CRF to improve the accuracy of semantic segmentation.

CRF inference with deep convolutional neural networks improves pixel-level label prediction by producing sharp boundaries and dense segmentation. Several methods learn arbitrary potentials in CRFs. It has been used as post-processing, end-to-end mode, formulated as RNN and integrated as a module into existing neural networks.

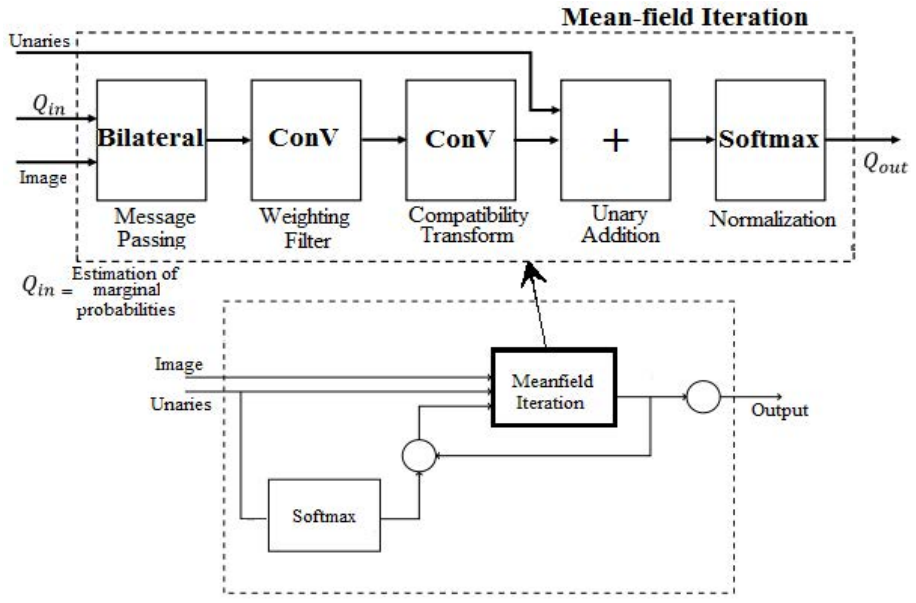


Figure 2.15: CRF as a recurrent Neural Network Zheng et al. (2015)

ALTERNATIVE TO CRF

Integrating the conditional random field into the original architecture is a difficult task due to the additional parameters and the high computational complexity of training. Moreover, the majority of CRFs use hand-constructed color-based affinities, which may lead to spatial false predictions. Several methods have been proposed to overcome these problems and can be used as an alternative to CRFs. **Table 2.12** shows network models that are an alternative to CRFs.

Table 2.12: Alternative to CRF based Methods

Category	Strategy / Structure	Corpus	Original Architecture	Testing Benchmark	Code Available
Alternative to CRF Approaches	Bilateral Neural Network (BNN) Jampani et al. (2016)	Bilateral filter inference in DenseCRF Replacing Gaussian potentials with bilateral convolution to learn pairwise potentials .	DeepLab	Pascal VOC	Yes
	Fast Bilateral Solver (BS) Barron and Poole (2016)	Edge-aware smoothness algorithm using bilateral filtering technique.	CRF-RNN	Pascal VOC MS COCO	-
	Boundary Neural Field (BNF) Bertasius et al. (2016)	Build unary and pairwise potentials from input RGB image, then combine them in global manner.	FCN	Semantic Boundaries Dataset	-
	DT-EdgeNet Chen et al. (2016a)	Domain transform (DT) Module: Edge-preserving filter. Edge Net: Predicts edge features from midway layers.	DeepLab	Pascal VOC	-
	Global Convolutional Network (GCN) Peng et al. (2017)	Large kernels used for classification and localization. Boundary Refinement Block: Model the boundary alignment as a residual structure.	FCN ResNet	Cityscapes COCO PASCAL VOC	-
	Boundary Refinement with Point Supervision BRPS Dong et al. (2021)	Boundary refinement module adopts the learned direction field to guide the object edge points rectification. Uncertainty estimation, key points detection and offset relaxation based on point supervised learning.	UNet	Cityscapes PASCAL VOC, NYUDv2, BDD100K	-
	Semantic Boundary Enhancement and Position network (SBEPNet) Chen et al. (2021b)	Boundary Enhancement Attention Module (BEAM) and Position Attention Module (PAM). Learn the long-range spatial inter dependencies along semantic boundaries to capture discriminative contextual information.	ResNet	Cityscapes, CamVid, PASCAL VOC	-
	Random Walk Network (RWN) Bertasius et al. (2017)	Random Walk Network Pixel labeling framework	DeepLab-largeFOV	Pascal, SBD-Stanford Background, Sift Flow	-

Bertasius et al. (2016) proposed an FCN architecture called Boundary Neural Field (BNF) for predicting semantic boundaries and building semantic segmentation maps using global optimization. The BNF combines the unary potentials (prediction by FCN) and the pairwise potentials (boundary-based pixel affinities) from the input RGB image in a global way. The basic idea is to assign pixels to foreground and background labels for each of the different object classes and apply constraint relaxation. Later in Bertasius et al. (2017) they proposed Convolutional Random Walk Network (RWN) which addresses the same problem, a model based on the random walk method Lovász et al. (1993). The

network model predicts semantic segmentation potentials and affinities at the pixel level and combines them through the proposed random walk layer that applies spatial smoothing predictions.

Jampani et al. (2016) developed a network based on a bilateral Gaussian filter Adams et al. (2010b) called bilateral neural network (BNN). Bilateral filter inference in fully connected CRF Krähenbühl and Koltun (2011) (by replacing Gaussian potentials with bilateral convolution) to learn pairwise potentials from fully connected CRF. Barron and Poole (2016) proposed an edge-aware smoothing algorithm using a bilateral filtering technique called the bilateral solver. Peng et al. (2017) proposed a residual based boundary refinement model, Global Convolutional network (GCN), for semantic segmentation. They proposed a Boundary Refinement Block (FCN structure without fully connected and global pooling layers) to model boundary alignment as a residual structure. Chen et al. (2016a) introduced a model with Domain Transform (DT) module as a replacement for CRF, an edge-preserving filtering method. The model consists of three modules. The first module generates a prediction of semantic segmentation results based on DeepLab. The second module named Edge Net predicts edge features from middle layers and the third module is an edge-preserving filter named Domain Transform (recursive filtering) proposed in Gastal and Oliveira (2011). The authors Chen et al. (2021b) introduced a Semantic Boundary Enhancement and position network (SBEPNet) that can detect semantic boundaries in a semantic segmentation task to improve high-level feature maps. The semantic boundaries can be efficiently obtained by explicitly exploiting the continuity of connected regions and overlaid with the original feature maps to improve the features. The Boundary Enhancement Attention Module (BEAM) is proposed to learn the long-range spatial dependencies along semantic boundaries to capture discriminative context information. Dong et al. (2021) present a lightweight boundary refinement module with point supervision named BRPS to improve the edge quality for the segmentation result produced by various existing segmentation models.

Several methods have been proposed that can be used as an alternative to CRF with the advantage of speed and fewer parameters. Bilateral filtering techniques can be a useful tool in the construction of deep learning frameworks.

The **Figure 2.16** gives the readers an overview of the categorization of the different semantic segmentation methods.

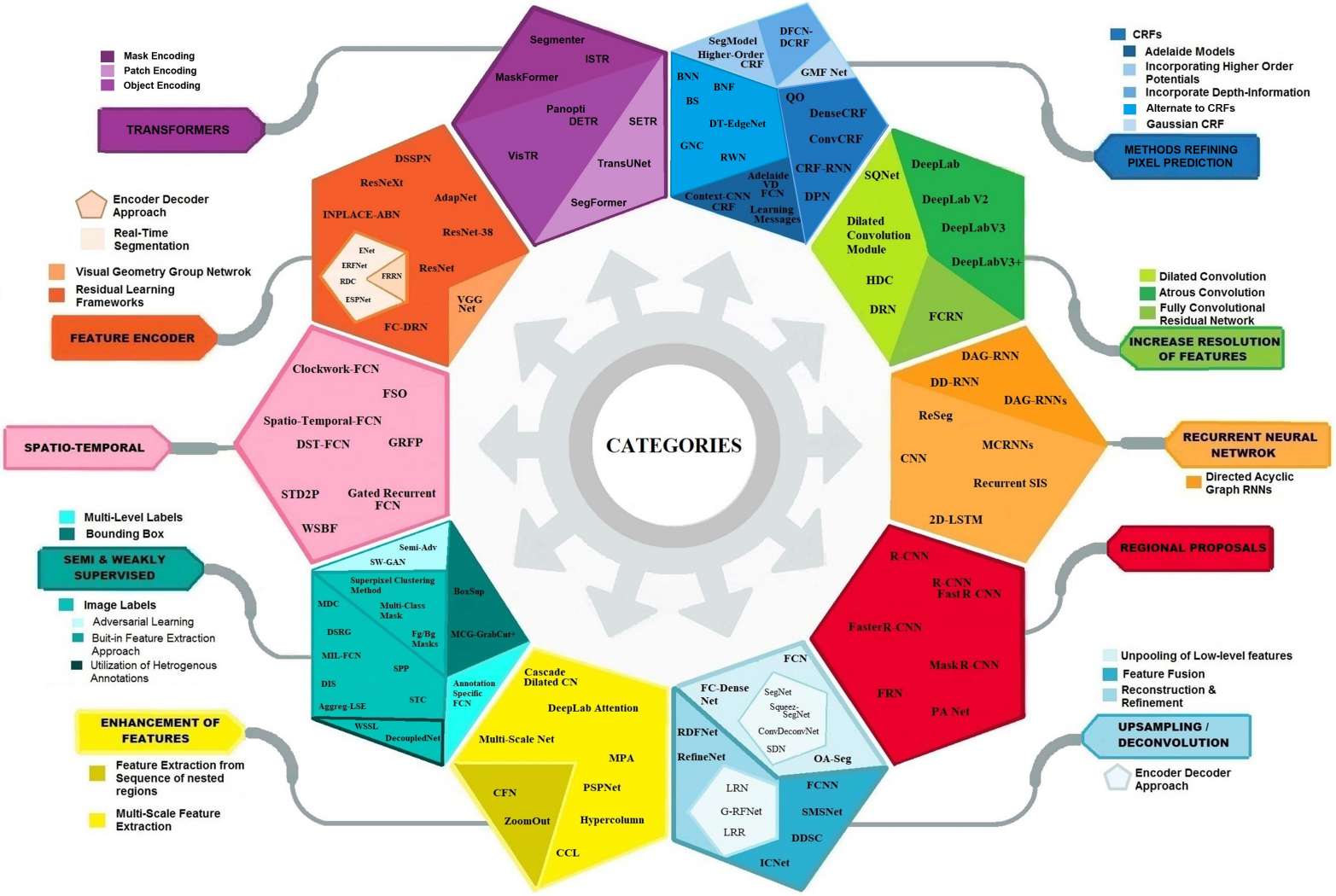


Figure 2.16: Illustration of the ten categories into which we have classified the reviewed semantic segmentation methods

2.2.2/ BENCHMARKS

One of the most difficult problems for all segmentation systems based on deep learning techniques is the collection of data to create a suitable dataset. There are four possible ways to obtain labeled data as shown in **Figure 2.17**. Traditional Supervision : Hand labeled data; Weak supervision: obtained automatically without human annotators using unlabeled data; Semi-supervised learning: partially labeled and partially unlabeled data and Transfer learning: using a pre-trained model as a starting point. The dataset serves as a benchmark against which deep learning networks are trained and tested. In recent years, several datasets have been created to be used in Deep Learning, motivating researchers to create new models and strategies with better generalization capabilities.

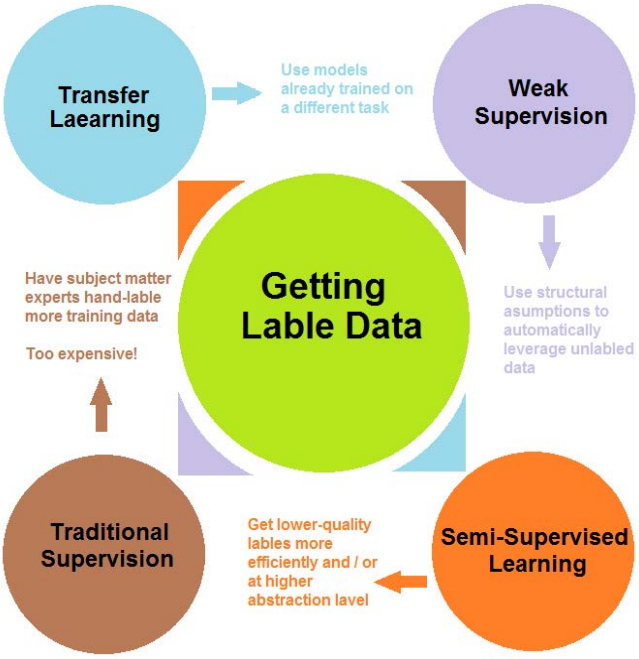


Figure 2.17: Getting Label Data

These datasets can be categorized according to the nature of data.

The automotive datasets include CamVid dataset Brostow et al. (2009), which is considered the first with semantically annotated videos, Daimler Urban Segmentation Scharwächter et al. (2013), CityScapes Cordts et al. (2015), Mapillary Vistas Neuhold et al. (2017) and the latest Apolloscape-Scene parsing Huang et al. (2018b), which focuses on semantic understanding of urban street scenes. The KITTI Geiger et al. (2012) dataset is used in various computer vision tasks such as 2D/3D object detection, stereo, optical flow and tracking. Synthetic datasets Ros et al. (2016a) Richter et al. (2016) consist of thousands of images extracted from realistic open-world games.

Table 2.13: **Summary of Datasets**

DataSet		Environment Nature	No of Classes	Samples *Training*	Samples *Validation*	Samples *Test*	Image Resolution	Year	Performance	Network Model
ADE20K Zhou et al. (2016b)]		Generic	150	20210	2000	-	Variable	2016	50.28% MIoU	SETR Zheng et al. (2021)
Apolloscape Scene parsing Huang et al. (2018b)		Street View / 2D-3D	25	146997 Frames			3384×2710	2018	33.9% MIoU	Megvii Li et al. (2020d)
Barcelona Tighe and Lazebnik (2010)		Outdoor	170	14871	-	279	640×480	2010	74.6% GL acc.	DAG-RNN Shuai et al. (2016)
BDD100K Yu et al. (2018)		Street View	40	70K	10K	20K	1280×720	2020	58.3% MIoU	BDDSNet Yu et al. (2018)
CamVid Brostow et al. (2009)		Street View	32		701		960×720	2009	82.9% MIoU	VPLR Zhu et al. (2019)
CIFAR-10/100 Krizhevsky and Hinton (2009)		Generic / Objects	10/100	50K/500	-	10K/100	32×32	2009	99.70 % correct	EffNet-L2 (SAM) Foret et al. (2020)
Cityscapes Armeni et al. (2017)	Fine	Street View	30	2975	500	1525	2048×1024	2016	84.9% MIoU	HMSA Tao et al. (2020)
	Coarse			22973	500	-			85.1% MIoU	HMSA Tao et al. (2020)
Cornell RGB-D Koppula et al. (2011)		Indoor Office/Home	-	24 Office / 28 Home Scenes Point Clouds			Variable	2011		
COCO StuffCaesar et al. (2016)		Generic	172	163957			Variable	2018	72.3% MAcc	IIC Ji et al. (2019)
DAVIS Pont-Tuset et al. (2017)		Generic / Videos	4	4219	2023	2180	480p	2017	90.4% MIoU	STCN Cheng et al. (2021b)
Data from Game Richter et al. (2016)		Synthetic / Street View	19	24966			1914×1052	2016	49.8% MIoU	MRKLD Yang et al. (2019b)
Daimler Urban Segmentation Scharwächter et al. (2013)		Street View / Video	5	500			1024×440	2013	77.2% MIoU	Layered Interpretation Liu et al. (2015a)
Freiburg Forest Valada et al. (2016)		Outdoor / Forest-Environment	6	230	-	136	1024×768	2016	88.25% MIoU	AdapNet Valada et al. (2017)
ImageNet Deng et al. (2009)		Generic	1 K	14,197,122			Variable	2010		
INRIA-Graz-02 Marszalek and Schmid (2007)		Outdoor /Natural	3	479	-	479	640×480	2007		
KITTI Geiger et al. (2012)		Street View	10	140	-	112	1226×370	2015	69.77% MIoU	SwiftNet Oršić andŠegvić (2021)
LabelMe Russell et al. (2008)		Outdoor	8	2920	-	1133	Variable	2008		
Mapillary Vistas Neuhold et al. (2017)		Street View	66	18000	2000	5000	1920×1080	2017	61.1% MIoU	HMSA Tao et al. (2020)
Microsoft COCO Lin et al. (2014a)		Generic	80	82783	40504	81434	Variable	2014	56.9% AP	FPN Lin et al. (2017c)
Microsoft Cambridge Shotton et al. (2011b)		Outdoor	21		591		320×240	2005		
NYUDv2 Silberman et al. (2012)		Indoor	40	795	654	-	480×640	2012	50.1% MIoU	RDFNet Park et al. (2017)
PASCAL	VOC Everingham et al. (2015)	Generic	20	1464	-	1449	Variable	2012	89.0% MIoU	DeepLabV3+ Chen et al. (2018)
	Context Mottaghi et al. (2014)	Generic	59	10103	-	9637	Variable	2014	55.8% MIoU	SETR Zheng et al. (2021)
	SBD Hariharan et al. (2011)	Outdoor	21	8498	-	2857	Variable	2011	82.1% MIoU	DeepLabv2+RWN Teichmann and Cipolla (2018)
RGB-D Object v2 Lai et al. (2011)		Household / Warehouse Objects	51	41877			640×480	2014		
ScanNetv2 Dai et al. (2017)		Indoor / 3D	20	+1500 scans			Variable	2018	57.7% MIoU	AdapNet++ Valada et al. (2019)
SegTrack v2 Li et al. (2013)		Generic / Videos	14	976 Frames			Variable	2013	80.12% MIoU	RFCNet Siam et al. (2017)
Sift-Flow Liu et al. (2011)		Outdoor	33	2488	-	200	256×256	2011	44.9% MIoU	Context-cNNLin et al. (2016)
Stanford	Background Gould et al. (2009)	Outdoor	8	715			320×240	2009	65.7% MIoU	MCRNN Fan et al. (2018)
	2D-3D Armeni et al. (2017)	Indoor / 2D-3D	13	70469 / 360° Scans			1080×1080	2017	49.9% fwIoU	Depth-CNN Wang and Neumann (2018)
SUN Dataset	3D Xiao et al. (2013)	Indoor / 3D / Video	-	19640 Frames			640×80	2013	58.5% IoU	LSTM-CF Li et al. (2016c)
	RGB-D [131]	Indoor / 2D-3D	37	2666	2619	5050	Variable	2015	48.1% MIoU	CCL Ding et al. (2018)
SYNTHIA Ros et al. (2016a)		Synthetic / Street View	11	13407			960×720	2016	92.1% MIoU	AdapNet++ Valada et al. (2019)
Synscapes		Synthetic / 3D Street View	30	25000			1440×720	2018	87.0% MIoU	DeepLabV3+ Chen et al. (2018)
Youtube Dataset Jain and Grauman (2014)		Objects / Video	10	+10000 Frames / 126 Videos			480×360	2014	68.5% MIoU	Clockwork-FCN Shelhamer et al. (2016)

Datasets generic in nature; PASCAL VOC Everingham et al. (2015) is one of the most popular and widely used datasets in the field of semantic segmentation by Deep Learning, CIFAR-10/100 Krizhevsky and Hinton (2009) contains up to 60,000 images providing 10 and 100 categories of tiny 32×32 images. A remarkable ImageNet Deng et al. (2009) dataset contains over 14 million labeled images, SegTrack v2 Li et al. (2013) is a video segmentation dataset with annotations to multiple objects at each frame, and PASCAL Context Mottaghi et al. (2014) is a set of additional annotations for PASCAL VOC. Microsoft- COCO Lin et al. (2014a) is a collection of images of complex everyday scenes with frequent natural objects, ADE20K Zhou et al. (2016b) contains both indoor and outdoor scenes with large variations, and DAVIS Pont-Tuset et al. (2017) is a dataset of densely annotated videos with pixel-precise ground truth. The recently developed COCO stuff Caesar et al. (2016) dataset extends the original COCO dataset with much richer stuff annotations.

Indoor environment datasets; NYUDv2 Silberman et al. (2012) consists of RGB-D images and video sequences from a variety of indoor scenes, Cornell RGB-D Koppula et al. (2011) contains labeled point clouds of office and home scenes, ScanNet Dai et al. (2017) includes more than 1500 scenes annotated with 3D camera pose, surface reconstructions, and semantic segmentation. Stanford 2D-3D Armeni et al. (2017) contains mutually registered modalities from 2D/3D domains, with 71,882 RGB images (both regular and 360°), along with corresponding depths, surface normals, and semantic annotations. SUN 3D Xiao et al. (2013) and SUN RGB-D Song et al. (2015) datasets include videos of large spaces for place-centric scene understanding.

Object datasets; RGB-D Object v2 Lai et al. (2011) contains 25000 images of common household items in 51 categories, YouTube dataset Jain and Grauman (2014) includes 126 videos.

Datasets for outdoor environment; Microsoft Cambridge Shotton et al. (2006) consists of 591 real photos of outdoor scenes with 21 object classes; Graz-02 Marszalek and Schmid (2007) is a dataset created at INRIA for object categories in nature scenes. LabelMe Russell et al. (2008) contains outdoor photos of 8 different classes taken in different cities in Spain; Barcelona dataset Tighe and Lazebnik (2010) is a subset of LabelMe; Stanford-background Gould et al. (2009) and PASCAL SBD Hariharan et al. (2011) are collected from PASCAL VOC; Sift-flow Liu et al. (2011) consists of 2688 images with 256×256 pixels and 33 classes, and Freiburg Forest Valada et al. (2016) depicts an outdoor forest environment under varying light, shade, and sun angle conditions.

Creating datasets is both time consuming and labor intensive, so for researchers and developers the most practical and viable approach is to use existing standard datasets that are representative enough of the domain of the problem. Some datasets have become standard and are often used by researchers to compare their work with others using stan-

dard metrics for evaluation. Selecting a dataset at the beginning of research is a difficult task, so providing a comprehensive description of the dataset can help.

In **Table 2.13**, the datasets used by deep learning networks that are publicly available are listed. Various information is provided, such as the type of environment, the number of classes, the training/test patterns, the image resolution, the year of construction, and the best performance obtained so far (to the best of our knowledge) by the semantic segmentation models. Shotton et al. (2011b); Koppula et al. (2011); Lai et al. (2011) Datasets are not used for semantics, but they can be used for semantic segmentation.

2.2.3/ EVALUATION METRICS

We describe commonly used evaluation metrics for semantic segmentation. The overall performance of semantic segmentation systems can be evaluated in terms of accuracy, time, memory, and power consumption.

Accuracy: The accuracy of the semantic segmentation system is a measure of the correctness of the segmentation, or is the ratio of the correctly segmented area to the ground truth.

Pixel wise Accuracy: The ratio between the amount of correctly classified pixels and the total number of pixels. Confusion matrix terminology is used to describe the performance of a classification model.

Let N_{cls} be the number of classes, N_{xy} the number of pixels belonging to class x and labeled as class y. The confusion matrix gives the number of false positives (N_{xy}), false negatives (N_{yx}), true positives (N_{xx}) and true negatives (N_{yy}).

$$PixelAccuracy = \frac{\Sigma_{x=1}^{N_{cls}} N_{xx}}{\Sigma_{x=1}^{N_{cls}} \Sigma_{y=1}^{N_{cls}} N_{xy}} \tag{2.1}$$

Pixel-wise classification accuracy is not reliable for the actual performance of a classifier, as it gives misleading results if the dataset is unbalanced (i.e., large regions that have a class or labeled images might have coarser labeling).

Mean Accuracy: The ratio of correct pixels is calculated per class and then averaged over the total number of classes N_{cls}.

$$MeanAccuracy = \frac{1}{N_{cls}} \Sigma_{x=1}^{N_{cls}} \frac{N_{xx}}{\Sigma_{y=1}^{N_{cls}} N_{xy}} \tag{2.2}$$

Mean Intersection over Union (MIoU): The ratio between the number of true positives N_{xx}, (Intersection) over the sum of true positives N_{xx}, false negatives N_{yx}, false positives

N_{xy} (Union). Intersection over union is calculated for each class and then averaged.

$$MIoU = \frac{1}{N_{cls}} \Sigma_{x=1}^{N_{cls}} \frac{N_{xx}}{\Sigma_{y=1}^{N_{cls}} N_{xy} + \Sigma_{y=1}^{N_{cls}} N_{yx} - N_{xx}} \tag{2.3}$$

The most widely used accuracy measuring strategy is MIoU, due to its easiness and simplicity.

Frequency Weighted Intersection over Union (FWIoU)

$$FWIoU = \frac{1}{\Sigma_{x=1}^{N_{cls}} \Sigma_{y=1}^{N_{cls}} N_{yx}} \Sigma_{x=1}^{N_{cls}} \frac{\Sigma_{y=1}^{N_{cls}} N_{xy} N_{xx}}{\Sigma_{y=1}^{N_{cls}} N_{xy} + \Sigma_{y=1}^{N_{cls}} N_{yx} - N_{xx}} \tag{2.4}$$

Precision: The relation between true positives N_{xx}, and all elements classified as positives

$$Precision = \frac{N_{xx}}{N_{xx} + N_{xy}} \tag{2.5}$$

Recall: measures how good all the positives are found.

$$Recall = \frac{N_{xx}}{N_{xx} + N_{yx}} \tag{2.6}$$

Average Precision: Mean precision at a set of eleven equal space recall levels (0.0, 0.1, 0.2 . . . , 1)

Mean Average Precision: Mean of all the Average Precision values across all classes.

Time, Memory and Power:

The memory and processing time of the system is highly dependent on the hardware and backend implementation. The use of hardware accelerator GPUs makes the processing time of these systems very fast, but consumes a lot of memory and power. Most of the methods do not provide information related to time, memory and hardware, which is very important because these network models can be used in areas (mobile systems, robotics, autonomous driving, etc.) where extremely accurate image segmentation is required with limited power and memory. Moreover, this information can help researchers to estimate, compare or select methods depending on the application and requirement.

2.2.4/ ANALYSIS

We analyze some of the network models based on their performance on datasets and their design structure to find out the reasons for their performances. It is difficult to compare these methods because most of them were evaluated on very few datasets. Some methods used different metrics and also lack information about the experimental setup

(hardware, time, memory).

AdapNet Valada et al. (2017):

- Achieves top score of 88.25% IoU on Freiburg Forest. The network reached Mean IoU of 69.39% on cityscapes and 72.91% on Synthia dataset.

The improvement is due to the highly representative multiscale features learned by the model, which allow segmentation of very distant objects present in Synthia and Cityscapes. AdapNet's modeling approach is based on a mixture of convolutional neural network (CNN) experts (Convoluted Mixture of Deep Experts - CMoDE) and considers multiple modalities such as appearance, depth and motion.

AdapNet++ Valada et al. (2019):

- Achieves top score of 92.1% IoU on Synthia and 57.7% IoU on ScanNetv2 dataset. The network achieves the score of 83.94% IoU on Cityscapes, 45.75% IoU on SUN RGB-D, and 84.18% IoU on Freiburg Forest dataset.

Self-Supervised Model Adaptation which includes a new encoder with multiscale residual units and an efficient atrous spatial pyramid pooling that has a larger effective receptive field with more than 10x fewer parameters, complemented by a strong decoder with a multi-resolution supervision scheme that recovers high-resolution details.
PSPNet Zhao et al. (2017):

- Competitive results are obtained on Cityscapes and Pascal VOC with 80.2% IoU and 85.4% IoU respectively.

PSPNet has developed an effective optimization strategy for Deep ResNet-101 He et al. (2016) based on deeply supervised loss; two loss functions: Main softmax loss to train the final classifier and auxiliary loss applied after the fourth stage, this helps in optimizing the learning process. PSPNet applies multi-scale tests, experiments with different depths of the pre-trained ResNet and performs data augmentation.

SETR Zheng et al. (2021):

- Achieves the best results on ADE20K and Pascal Context with 50.28% IoU and 55.83% IoU respectively. Promising results are obtained on cityscapes with 80.2% IoU.

SEgmentation TRansformer (SETR) is an encoder-decoder based network model. In SETR encoder, the stacked convolution layers with gradually reduced spatial resolution are replaced by a pure transformer Vaswani et al. (2017). This pure transformer encoder treats an input image as a sequence of image patches represented by learned

patch embedding, and transforms the sequence with global self-attention modelling for discriminative feature representation learning. The authors also proposed three different designs for the decoder: Naive Upsampling (Naive), Progressive Upsampling (PUP) and Multi-Level feature Aggregation (MLA). The model achieves the best results with the MLA assumption.

FCCN Yang et al. (2019a):

- Achieves a scores of 69.94% IoU on CamVid and score of 44.23% IoU on ADE20K dataset.

FCCN proposed a cost function that significantly improves segmentation performance. Very few researchers attempted to modify the cost function when training their models. FCCN computes the cost function on each pre-output layer including the final output layer.

VPLR Zhu et al. (2019):

- Achieves a top score of 82.9% IoU on CamVid, and a score of 83.5% IoU on Cityscapes dataset.

A joint propagation strategy is proposed to mitigate misalignment's in synthesized patterns. The training segmentation models on datasets augmented with the synthesized samples leads to significant improvements in accuracy. The novel boundary-label relaxation technique makes training robust to annotation noise and propagation artifacts along object boundaries.

DeepLab V3 Chen et al. (2017b):

- Achieves score of 81.3% IoU on cityscapes.

The improvement comes mainly from changing the hyper-perimeter: fine-tuning batch normalization, varying batch size, larger clipping size, changing the output stride, multi-scale inputs during inference, adding left-right flipped inputs, trained on 3475 finely and additional 20000 coarsely annotated images of the Cityscapes dataset. Furthermore, using the ResNet-101 model pre-trained on ImageNet and the JFT dataset yields the second best score of 86.90 IoU on Pascal VOC.

DeepLab V3+ Chen et al. (2018):

- Achieves 89.0% IoU on Pascal VOC and 82.1% IoU on cityscapes.

DeepLab V3+ is a modified version of DeepLab V3, adapted to output stride = 16 or 8 instead of 32. It is also adapted to the Xception module, further increasing performance.

DSSPN Liang et al. (2018):

- Achieves score of 38.9% IoU on COCO, 43.6% IoU on ADE20K, 58.6% IoU on Pascal Context and 45.01% IoU on Mapillary dataset.

DSSPN constructs a semantic neuron graph in which each neuron segments regions of a parent concept in a semantic concept hierarchy (by combining labels from four datasets) and aims to recognize between its child concepts. Instead of using a completely large semantic neural graph, DSSPN only activates a relatively small neural graph for each image during training, making DSSPN memory and computationally efficient.

RFCNet Siam et al. (2017):

- Achieves scores of 81.20% IoU on SYNTHIA, 80.12% IoU on SegTrack and competitive score of 69.84% IoU on DAVIS dataset.

The model uses different FCN architectures such as a recurrent node to use temporal information, a deconvolution layer for upsampling, and a support skip architecture for finer segmentation. The use of temporal data is the reason for the performance improvement and not the simple addition of extra convolutional filters.

Adelaide Context CNN-CRF Lin et al. (2016):

- Achieves score of 40.6% IoU on NYUDv2, 42.30% IoU on SUN-RGB, 78.00% IoU on Pascal VOC, 66.40% IoU on CIFAR-100, 71.60% IoU on Cityscapes, and 43.30% IoU on Pascal Context dataset.

The model uses CNN-based pairwise potential functions to capture semantic correlations between neighboring patches that improve coarse-level prediction. The model uses FCN with sliding pyramid pooling, CNN contextual pairwise, boundary refinement (dense CRF method) and trained the model with additional images from the COCO dataset to improve the overall performance of the model.

Clockwork-FCN Shelhamer et al. (2016):

- Achieves 68.50% IoU on Youtube Object, 68.40% IoU on Cityscapes, 28.90% IoU on NYUDv2 dataset.

Clockwork-FCN uses different clocking schemes; fixed-rate clock reduces computational overhead by assigning different clock rates to each stage, so that later stages execute less often. Adaptive clockwork updates when the output score maps are expected to change, reducing computation while maintaining accuracy.

SwiftNet Oršić andŠegvić (2021):

- Achieves top score of 69.77% IoU on KITTI, and reach 76.4% mIoU on Cityscapes. Further, 55.0% IoU on Camvid and 44.8% IoU on Mapillary dataset.

Within SwiftNet, two approaches to increasing the size of the receptive field are considered. First, Spatial Pyramid Pooling (generates feature maps with varying levels of detail by enriching features from the encoder output with their pools over coarse spatial grids 1×1, 2×2, 4×4, and 8×8.) Second, Pyramidal Fusion (true multiscale representations, train with the boundary-aware loss to avoid overfitting). This shows significant improvements on all tested datasets.

Residual framework ResNet-38 Wu et al. (2019b):

- Achieves the highest score of 48.1% IoU on Pascal Context, 80.6% IoU on cityscapes and 43.43% IoU on ADE20K.

The model introduces residual units into ResNet (17 residual units for 101 layers of ResNet) and extends it into a sufficiently large number of subnets. Each connection in the ResNet unit shares same kernel size and number of channels, which improves model accuracy. ResNet-38 does not apply multi-scale testing, model averaging, or CRF-based post-processing, except for the ADE20K test set.

ESPNet: Mehta et al. (2018):

- Efficient real-time segmentation network, achieves 60.2% IoU on cityscape, 40.0% IoU on Mapillary dataset with 0.364M parameters, 63.01% IoU on Pascal VOC test set with 0.364M parameters.

Efficient Spatial Pyramid (ESP) network is an efficient neural network in terms of speed and memory. ESP , based on factorized form of convolutions (pointwise convolution and spatial pyramid of dilated convolutions), reduces the number of parameters, memory, with large receptive field.

FCN-8s Long et al. (2015):

- Achieves the score of 77.46% IoU on Freiburg Forest, 67.20% IoU on PASCAL VOC, 65.30% IoU on CIFAR-10, 65.30% IoU on Cityscapes, 56.10% IoU on KITTI, 29.39% IoU on ADE20K, 35.10% IoU on PASCAL CONTEXT, 65.24% IoU on SYNTHIA, and 57.00% IoU on CamVid dataset.

Performance is enhanced by transferring pre-trained classifier weights, fusing different layer representations, and learning on whole images throughout.

DAG-RNN Shuai et al. (2017):

- Achieves 44.8% IoU on Sift-flow, 31.2% IoU on COCO (171 classes) and 43.7% IoU on PASCAL Context dataset.

The segmentation network uses a pre-trained CNN with DAG -RNN that fuses low-level features with DAG -RNN. A new class-weighted loss function is proposed to control the class-wise loss during training. The performance of the segmentation network increases with increase of DAGs with DAG -RNN. A fully connected CRF is used to further improve the performance of the network.

RefineNet Lin et al. (2017b):

- Achieves a score of 45.90% IoU on SUN-RGB, 46.50% IoU on NYUDv2 and 47.30% IoU on Pascal Context datasets. The results on Pascal VOC, cityscapes, and ADE20K datasets are 83.40% IoU, 73.60% IoU, and 40.70 % IoU respectively.

RefineNet applies data augmentation during training (random scaling, cropping, and horizontal flipping of the image) and multiscale evaluation (averaging predictions for the same image over different scales for the final prediction). The Dense CRF method is only used for Pascal VOC.

Dilation10 Yu and Koltun (2015):

- Achieves 67.60% IoU on PASCAL VOC, 67.10% IoU on Cityscapes, 32.31% IoU on ADE20K and 65.29% IoU on CamVid dataset.

The model is an adapted version of Shuai et al. (2016), where the pooling and convolutional layers of conv4/conv5 are replaced by two dilated convolutional layers with dilation factors of 2 and 4, respectively. This leads to a reduction in the size of the network and its runtime for real-time applications.

ResNet DUC+HDC Wang et al. (2018):

- Achieves a score of 80.10% IoU on Cityscapes, 83.10% IoU on PASCAL VOC, 39.40% IoU on ADE20K dataset.

DUC provides the dense pixel-wise predictions, HDC uses arbitrary dilation rates that increase the receptive fields of the network. Experiments are performed using ResNet at different depths, and data augmentation is applied (for cityscapes, each image in the training set is partitioned into twelve 800×800 patches, yielding 35700 images). The model is trained using the combination of the MS - COCO dataset, augmented PASCAL VOC 2012 training set, and the valid training set. ResNet DUC +HDC is also evaluated on the KITTI dataset and achieves an average precision of 92.88% for road segmentation using the ResNet 101- DUC model pre-trained from ImageNet during training.

HMSA Tao et al. (2020):

- Achieves top scores of 61.1% IoU on Mapillary Vistas, and 85.1% IoU on Cityscapes dataset.

Hierarchical multiscale attention mechanism by which the network learns to predict the relative weights between adjacent scales. This requires only the addition of one additional scale to the training pipeline, whereas SOTA methods require each additional inference scale to be explicitly added during the training phase. A hard threshold based auto-labelling strategy that uses unlabeled images and improves IOU.

ST-Dilation Fayyaz et al. (2016):

- Achieves the score of 65.90% IoU on CamVid dataset. Model ST-FCN32s scores 50.60% IoU on Camvid dataset and Model ST-FCN8s scores 30.90% IoU on NYUDv2 dataset.

No post-processing is required in the STFCN model, the spatio-temporal module is embedded on the last convolutional layer. LSTM blocks are used to derive the relationships between spatial features, which provide valuable information and improve the accuracy of segmentation. Moreover, the application of dilated convolutions for contextual information at multiple layers leads to better results.

STGRU (GRFP + Dilation) Nilsson and Sminchisescu (2016):

- Achieves the score of 66.10 IoU on CamVid dataset. Model GRFP + Dilation scores 67.80% IoU and model GRFP + LRR-4x achieves the score of 72.80% IoU on Cityscapes dataset.

The model combines the power of both convolutional-gated architecture and spatial transformers (CNN). The model GRFP is trained with Dilation 10 [88] and LRR [70] network which improve the performance for video. The model improves semantic video segmentation and labeling accuracy by propagating information from labeled video frames to nearby unlabeled frames with low computational overhead.
It can be noted that the methods that achieve the high performance results do so because of the availability of a large amount of labelled data. Additional training data is beneficial to increase the accuracy of the model; several models used large datasets (merging two or three datasets) when training.

Table 2.14: Links to the Source Codes

Network Model	Code Link	Network Model	Code Link
InceptionSzegedy et al. (2015)	https://github.com/Microsoft/CNTK/tree/master/examples/Image/Classification/GoogLeNet	RSIS Salvador et al. (2017)	https://github.com/imatge-upc/rsis
BN-Inception Ioffe and Szegedy (2015)		FC-DenseNet Jégou et al. (2017)	https://github.com/SimJeg/FC-DenseNet
Inception V2, V3 Szegedy et al. (2016)		ConvDeconvNet Noh et al. (2015)	https://github.com/HyeonwooNoh/DeconvNet
Inception V4 Szegedy et al. (2017)	https://github.com/titu1994/Inception-v4	SegNet Badrinarayanan et al. (2017)	https://github.com/alexgkendall/caffe-segnet
Xception Chollet (2017)	https://github.com/kwotsin/TensorFlow-Xception	FCN Long et al. (2015)	https://github.com/shelhamer/fcn.berkeleyvision.org
VGGNet Simonyan and Zisserman (2014)	https://github.com/machrisaa/tensorflow-vgg	SMSNet Vertens et al. (2017)	https://github.com/JohanVer/SMSnet
ResNet He et al. (2016)	https://github.com/KaimingHe/deep-residual-networks	ICNet Zhao et al. (2018)	https://github.com/hszhao/ICNet
ResNet-38 Wu et al. (2019b)	https://github.com/itijyou/ademxapp	RefineNet Lin et al. (2017b)	https://github.com/guosheng/refinenet
ResNeXt Xie et al. (2017)	https://github.com/facebookresearch/ResNeXt	RDFNET Park et al. (2017)	https://github.com/SeongjinPark/RDFNet/blob/master
INPLACE-ABN Bulo et al. (2018)	https://github.com/mapillary/inplace_abn	G-FRNet Amirul Islam et al. (2017)	https://github.com/mrochan/gfrnet
FRRN Pohlen et al. (2017)	https://github.com/TobyPDE/FRRN	LRN Islam et al. (2017)	https://github.com/golnazghiasi/LRR
ENet Paszke et al. (2016)	https://github.com/TimoSaemann/ENet	DeepLab Chen et al. (2014)	https://bitbucket.org/deeplab/deeplab-public
ERFNet Romera et al. (2017)	https://github.com/Eromera/erfnet	DeepLabV2 Chen et al. (2017a)	https://bitbucket.org/aquariusjay/deeplab-public-ver2
ESPNet Mehta et al. (2018)	https://github.com/sacmehta/ESPNet	Dilation Yu and Koltun (2015)	https://github.com/tensorflow/models/tree/master/research/deeplab
R-CNN Girshick et al. (2014)	https://github.com/rbgirshick/rcnn	DeepLabV3+ Chen et al. (2018)	https://github.com/fyu/dilation
Fast R-CNN Girshick (2015)	https://github.com/rbgirshick/fast-rcnn	HDC Wang et al. (2018)	https://github.com/TuSimple/TuSimple-DUC
Faster R-CNN Ren et al. (2015)	https://github.com/ShaoqingRen/faster_rcnn	DRN Yu et al. (2017)	https://github.com/fyu/drn
Mask R-CNN He et al. (2017a)	https://github.com/matterport/Mask_RCNN	PSPNet Zhao et al. (2017)	https://github.com/hszhao/PSPNet
FPN Liu et al. (2018)	https://github.com/unsky/FPN	DenseCRF Krähenbühl and Koltun (2011)	https://github.com/lucasb-eyer/pydensecrf
DecoupledNet Hong et al. (2015)	https://github.com/HyeonwooNoh/DecoupledNet	GCRF Vemulapalli et al. (2016)	https://github.com/siddharthachandra/gcrf
WSSL Papandreou et al. (2015)	https://bitbucket.org/deeplab/deeplab-public	ConvCRF Teichmann and Cipolla (2018)	https://github.com/MarvinTeichmann/ConvCRF
Semi-Adv Hung et al. (2018)	https://github.com/hfslyc/AdvSemiSeg	BNN Jampani et al. (2016)	https://github.com/MPI-IS/bilateralNN
DSRG Huang et al. (2018c)	https://github.com/speedinghzl/DSRG	Clockwork Shelhamer et al. (2016)	https://github.com/shelhamer/clockwork-fcn
MCG GrabCut+ Khoreva et al. (2017)	https://github.com/philferriere/tfwss	STFCN Fayyaz et al. (2016)	https://github.com/MohsenFayyaz89/STFCN
ReSeg Visin et al. (2016)	https://github.com/fvisin/reseg	FSO Kundu et al. (2016)	https://bitbucket.org/infinitei/videoparsing

2.2.5/ Open Problems and Possible Solutions

1. **Reducing Complexity & Computation:**
   Deep neural networks are not very suitable for use on mobile platforms (e.g., embedded devices), which have limited resources, because DNNs are memory-intensive, time-consuming, and energy-consuming. There is also a problem with computational complexity due to a large number of operations required for inference. It is important to investigate how to reduce the complexity of the model to achieve high efficiency without loss of accuracy. Some CNN compression approaches have been proposed to reduce the complexity and computational cost. Wang et al. Wang et al. (2017) proposed a method to remove and reduce the redundancy in feature maps extracted from a large number of filters in each layer of the network. Kim et al. Kim et al. (2015) proposed a one-shot approach to compress the entire network consisting of three steps: rank selection, low-rank tensor decomposition, and fine-tuning. Andrew et al. Holliday et al. (2017) applied model compression techniques to the problem of semantic segmentation. Caffe2 is a portable deep learning framework from Facebook that is capable of training large models, and allows machine learning applications to be built for mobile systems. DNN compression and acceleration has made a lot of progress. However, there are some potential problems such as: Compression may lead to accuracy loss; Decomposition process; Transfer of information to convolutional filters is not suitable for some networks.

2. **Apply to Adverse Conditions:**
   There are a few network models that are used in real-world, challenging environments or deal with adverse conditions such as direct lighting, reflections from reflective surfaces, changing seasons, fog, or rain. Although some CNN models have used synthetic data along with real data to enhance the performance of state-of-the-art methods for semantic segmentation under challenging environmental conditions. However, the use of large amounts of high-quality real-world data is still indispensable so far. One possible solution is to use synthetic data together with real-world data. Obviously, there are significant visual differences between the two data domains and to reduce this gap, a domain adaptation technique can be used. Hoffman et al. Hoffman et al. (2016) proposed an unsupervised domain adaptation method to transfer semantic segmentation FCNs across image domains. Yang et al. Zhang et al. (2017) proposed a curriculum-like learning approach to minimise the domain gap. The authors in Sankaranarayanan et al. (2018) proposed a domain shift approach based on Generative Adversarial Network (GAN), which transfers the target distribution information to the learned embedding using a

generator-discriminator pair.

3. **Need large and high quality labeled data:**
   The classification performance of DNNs and dataset size are positively correlated. Current state-of-the-art methods require high quality labeled data, which is not available on large-scale as they are time consuming and labour exhaustive. The effective solution to this problem would be to build large and high quality datasets, which seems hard to achieve. Therefore, the researchers rely on semi and weakly supervised methods making DNNs less reliant on the labeling of large datasets. These methods has considerably improved the semantic segmentation performance by using additional weak annotations either alone or in combination with a small number of strong annotations. However, they are far from fully supervised learning methods in terms of accuracy. Thus, this opens new challenges for improvement.

4. **Overfitting:**
   As mentioned earlier, DNNs are data hungry and do not perform well unless fed with large datasets. The majority of available datasets are relatively small, so DNN models become very complex to capture all the useful information needed to solve a problem. With a limited amount of data, there is a risk of ”overfitting” the model. Overfitting occurs when the gap between the training error and the test error is too large. Regularization techniques help to overcome this problem. Regularization is any modification we make to a learning algorithm that aims to reduce its generalization error but not its training error Goodfellow et al. (2016a). Several of these methods are applied in DNNs to prevent overfitting, e.g., L1 and L2 regularization, Lp norm, dropout, DropConnect. Data Augmentation is also used to reduce overfitting (e.g., increase training data size - rotate, flip, scale, move images). However, regularization can increase training time (e.g., using dropout increases training time by 2 or 3 times compared to a standard neural network of the same architecture) and there is no standard for regularizing CNNs. Introducing a better or improved regularization method would be an interesting direction for future work.

5. **Segmentation in Real-time:**
   Real-time semantic segmentation without losing too much accuracy is of great importance as it can be useful in autonomous driving, robot interaction, and mobile computing where runtime is crucial to evaluate system performance. DNN methods for semantic segmentation are more focused on accuracy than speed.

The majority of methods are far from real-time segmentation. One possible solution to the problem could be to perform convolutions in an efficient way. Several works aim to develop efficient architectures that can run in real time and are based on convolution factorization (decomposition of the convolution operation into several steps). Some computationally efficient modules for convolution have been presented. For example, Inception Szegedy et al. (2015), Xception Chollet (2017), ResNet He et al. (2016), ASP Chen et al. (2017a), ESP Mehta et al. (2018); ShuffleNet Ma et al. (2018) and MobileNet Howard et al. (2017), use grouped and depthwise convolutions. Another possible solution would be to apply network compression using various techniques (e.g., parameter pruning and sharing Li et al. (2016a), low-rank factorization and sparsity Jaderberg et al. (2014), etc.) to reduce the size of the network. However, real-time semantic segmentation still lacks higher accuracy, and new methods and approaches need to be developed to find a trade-off between runtime and accuracy.

6. **Video / 3D Segmentation:**
DNNs have been successfully used for semantic segmentation of 2D images, while they are hardly used for 3D images and on videos despite their importance. Several video and 3D network models for semantic segmentation have been proposed over the years and progress has been made, but there are still some challenges. The lack of large datasets of 3D images and sequence images (videos) makes it difficult to make progress in semantic segmentation of 3D and video images. 3D networks are computationally expensive when dealing with high resolution and complex scenes (large number of classes). In the task of 3D semantic segmentation, the use of 3D point cloud information is very effective. Zhang et al. Zhang et al. (2018a) proposed an efficient large-scale point cloud segmentation method by fusing 2D images with 3D point clouds to CNN to segment complex 3D urban scenes. The authors in Yousefhussien et al. (2018); Charles et al. (2017) proposed methods for direct semantic labeling of 3D point clouds with spectral information. However, 3D segmentation methods face many challenges compared to 2D segmentation, i.e., high complexity, computational cost, slow processing, and most importantly, a lack of 3D datasets. In semantic video segmentation, two approaches can be useful, one to improve the computational cost (by reducing the latency); The authors in Shelhamer et al. (2016); Li et al. (2018) proposed designed scheduling frameworks that reduce the overall cost and maximum latency of semantic video segmentation. However, these approaches are far from meeting the latency requirements in real-time applications. The second approach is to improve accuracy (by exploiting temporal continuity - temporal features and temporal correlations between video frames). Several methods Fayyaz et al. (2016); He et al. (2017b); Qiu et al. (2018)

have been proposed that use temporal information with spatial information to increase the accuracy of pixel labeling.

2.3/ CONCLUSION

In this chapter, a comprehensive overview of deep learning techniques used for semantic segmentation has been given. The methods reviewed have been categorized into ten classes according to the common concept underlying their architectures. A summary of these methods was also provided, indicating for each method the main idea, the origin of its architecture, test benchmarks, code availability (Table 2.14 provides links of available source codes) and year of publication. Thirty-five datasets to which these methods were applied were reported and described in detail, indicating the type of environment, number of classes, resolution, number of images, and the method that, to the best of our knowledge, achieved the best performance on each dataset. We mainly analyzed the design and performance of some of these methods that were reported to have achieved high scores. The goal was to find out how they do this. We also discussed some of the open problems and tried to suggest some of the possible solutions. The study showed that there is a lot of room for improvement in terms of accuracy, speed and complexity.

3

Visual Attention for Urban Driving

3.1/ Introduction

3.1.1/ Problem Statement and Motivation

Autonomous driving is a challenging problem that requires a complete understanding of the visual environment. Predicting or locating potential risks and understanding the driving environment in the presence of discriminative properties such as ”darting-out pedestrian on a busy road, approaching vehicles, traffic light changes, or other traffic dynamics” is a skill that humans possess. Their sensory system allows them to quickly locate objects of interest, processing only the important details and ignoring the unnecessary within the scene. But how should a machine-learning system or autonomous vehicle acquire this ability to recognize such attentions for safe driving?

Numerous approaches have been proposed to address this problem by incorporating the ***saliency mechanism*** as a ***visual attention*** model. These models measure the salience of a location or the likelihood that a location will attract a human driver's attention (e.g., eye gaze, depth-of-field effect, road and traffic sign detection, etc). During the driving task, the environment changes dynamically over time and it is critical to focus attention on multiple objects simultaneously, **Figure 3.1** is an example of this. Row 1 of the example shows that the driver must pay attention to both the pedestrian and the traffic light simultaneously. The green light signals ”good to go,” but the driver must wait for the pedestrian to cross the street to avoid a collision. Similar situation in row 2, the darting pedestrian with his dog crossing the road without a crosswalk. The driver has to pay attention to him as well as to vehicles and traffic lights at the same time.

Previous research in cognitive studies recognizes that visual attention is object-based rather than location-based and that it varies with object motion Duncan (1984) O'Craven

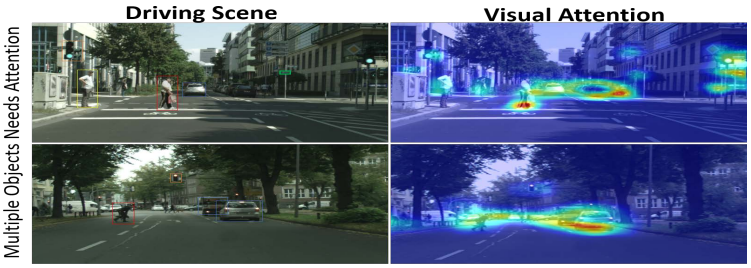


Figure 3.1: Example of Visual Attention for Driving

et al. (1999) Sears and Pylyshyn (2000). Most attentional models for driving incorporate human eye tracking into the process, where the driver sits with an eye tracker that records fixations (gaze areas or targets). Research shows that these models have contributed a lot to the use of attention and represent a significant advance. However, these models have some drawbacks, such as that they still suffer from the complexity of capturing the driver's actual attention. The fixations of different drivers vary on the same scene, which could lead to false gaze as they are subjected to different characteristics of the driver, i.e., driving experience & habits, preferences & intentions, abilities, culture & environment, age, gender, etc. Moreover, at each moment the driver looks at the vehicle, the eye tracker records only a single location, while he may look at several important objects in the scene.

Given the above statements, we came up with a novel idea by shifting the problem from prediction (*Where the driver looks at or where most drivers would look at*) to selection (*what the driver should/must look at*) while driving using a generative adversarial network, an approach to generative modeling using Deep Learning. Capturing what the driver would or should look at. It is important to first identify the important objects. Then the type of object is identified, i.e. vehicle, bicycle, cyclist, etc. It is also important to determine the location and movement of these objects and to be able to estimate the distance and direction of movement, i.e. whether each object detected has the potential to become a hazard to the vehicle. In this work, we will try to solve the first part of the question to detect important objects. We first review well-known saliency algorithms, both classical and deep learning, used for visual attention and evaluate their applicability to the driving environment. Followed by our new approach to visual attention for driving based on conditional Generative Adversarial Network. Then, We present our new strategy for obtaining data saliency heatmaps from existing publicly available datasets.

3.2/ Related Works

Modeling visual attention is an active research topic in image processing and computer vision, and is closely related to topics such as object saliency detection and gaze fixa-

tion. Our review focuses on saliency detection models, both classical and deep-learning based, used for visual attention in general and in the driving environment in particular.

3.2.1/ VISUAL ATTENTION USING CLASSICAL APPROACH

The term visual attention was used early in "Feature Integration Theory" by Treisman and Gelade (1980) to define human visual search strategies. According to this theory, salient areas in the visual scene are identified by the combination or relationship of visual feature information such as color, orientation, spatial frequency, brightness, direction of motion that direct human attention. The concept of saliency map was first proposed by Koch and Ullman (1987) to achieve attentional selection according to Treisman theory Treisman and Gelade (1980). The visual attention methods that use saliency are divided into two categories: **bottom-up** (biologically inspired methods; image color and intensity are common examples) and **top-down** (true computational methods; prior knowledge, memories, goals are common factors). Itti and Koch proposed a visual attention mechanism Itti et al. (1998) inspired by Treisman and Gelade (1980) and Koch and Ullman (1987). Their saliency detection model extracts multi-scale image features by covering different size ratios between the center and surrounding regions and combining them into a single saliency map. This classical model is considered one of the successful and widely used methods for selective attention in the human visual system. Based on its success, Harel et al. (2007) proposed a model called Graph-based visual saliency (**GBVS**), which applies the graph algorithms to achieve efficient saliency computations. Hou and Zhang (2007) makes use of the spectral residuals approach. The model, called Spectral Residual Model (**SR**), is based on the logarithmic spectral representation of images.

Frintrop (2006) introduced a new attention system called Visual Object detection with Computational Attention System - **VOCUS** that detects regions that are more likely to contain relevant information in the image (region of interest). Hou et al. (2011) proposed an algorithm called **SignatureSal**, which is a comprehensive image descriptor that detects salient regions in the image. An efficient saliency detection algorithm called **BMS** proposed by Zhang and Sclaroff (2013) is based on a set of random thresholded Boolean maps. A new form of VOCUS saliency method called **VOCUS 2** is proposed by Frintrop et al. (2015). The idea is to measure the center-surround contrast at different scales (Gaussian difference), and the model provides pixel-precise saliency maps. A similar center-surround difference logic is used in Montabone and Soto (2010), which proposed a "fine-grained" saliency algorithm called Visual Saliency Feature (**VSF**). The algorithm provides fine-grained feature maps and much better defined boundaries. Zhang and Zhou (2018) presents a saliency detection framework that uses object proposals in an unsupervised manner. Few attention methods are based on Multiple Object Tracking Theory (MOT) Pylyshyn and Storm (1988). The theory is that each object in the visual field has a

priority value for attention that is assigned in a goal-directed manner. Objects are indexed to this value and quickly attended to before other objects. Lee et al. (2008) proposes a visual attention model that finds out an object or set of objects that could possibly receive more attention from the user without considering the position of the viewpoint. Therefore, different methods with different assumptions and predictions have been developed for modeling attention.

3.2.2/ VISUAL ATTENTION USING DEEP LEARNING

A new wave of developments and improvements in saliency or attention prediction has been observed through the use of deep learning architectures. Provided with enough training data, these architectures have performed well.

Vig et al. (2014) proposed the **eDN** (ensembles of deep networks) saliency prediction model, learns complex and plausible salient features from gaze-labeled natural images. The eDN model performs better than the **DeepGaze** Kümmerer et al. (2014), the first model that used transfer learning for saliency prediction. The DeepGaze model was first an end-to-end deep convolutional neural network for the saliency prediction task using Alexnet. Later, they built **DeepGaze II** saliency model Kummerer et al. (2017) based on VGGNet, which uses a pointwise nonlinear combination of deep features. Another deep learning framework **SalDet**, which combines global and local context in a multi-context system for saliency detection, was proposed in Zhao et al. (2015). Saliency in Context (**SALICON**) Huang et al. (2015) is a selective visual attention model that incorporates information at multiple scales to predict human fixations. Models such as **ML -Net** Cornia et al. (2016) learn hierarchies of visual features extracted by CNN to predict saliency. The saliency detection framework in Jia et al. (2016) is based on two models, a generative model that measures saliency through sparse residuals based on the background dictionary, and a discriminative model that distinguishes objects from the background using neighbourhood information. **DeepFix** Kruthiventi et al. (2017) network architecture was developed to capture object-level semantics at different scales and extract local/global features for predicting eye fixations and salient objects in the image. Tavakoli et al. (2017) presents the saliency prediction algorithm **iSEEL** based on similarities between images and an ensemble architecture (deep convolutional neural networks) that constructs saliency maps. Wang et al. (2019) presents a pyramid attentive and salient edge-aware saliency model called **PAGE -Net**. The authors proposed a salient edge detection module that emphasises the importance of salient edge information as it provides a strong hint for better segmentation of salient objects and refinement of object boundaries. Hsu et al. (2019) proposed a weakly supervised method for top-down saliency detection, where the idea is to focus on the regions of specific objects that indicate the presence or absence of a target object in an image.

Some saliency models use the recurrent neural network as an attentional mechanism. Kuen et al. (2016) proposed a recurrent attentional convolutional-deconvolutional network (**RACDNN**) that continuously selects local regions and progressively refines the saliency prediction of these regions. Recurrent Mixture Density Network (**RMDN**) Bazzani et al. (2016) is a visual attention model that learns from human fixation data. Cornia et al. (2018) proposed a recurrent attention model called Saliency Attentive Model (**SAM**), which combines the power of a recurrent convolutional network and a fully convolutional network. The Deep Spatial Contextual Long Term Recurrent Convolutional Network (**DSCLRCN**) proposed by Liu and Han (2018) incorporates global and scene contexts to determine image saliency.

In recent years, researchers have shown the potential application of a generative adversarial network (GAN) Goodfellow et al. (2014b) for saliency detection of images. Several GAN based saliency detection methods have been proposed to generate synthetic saliency maps. Pan et al. (2017) proposed a method called **SalGAN** based on convolutional encoder-decoder architecture. It consists of two networks, a generator network trained with binary cross entropy (BCE) on existing saliency maps, and a discriminator network that identifies whether the given saliency map was created from actual fixations or by the generator. A fully supervised saliency detection model Supervised Adversarial Network (**SAN**) is proposed by Pan and Jiang (2017). Zhu et al. (2018) proposed a multi-scale adversarial feature learning model (**MAFL**) for image saliency detection. DSAL-GAN Mukherjee et al. (2019) was developed for salient object detection in noisy images. The model uses cycle consistency loss to refine saliency. Recently, Che et al. (2019) proposed the saliency model **GazeGAN**, which incorporates skip connections (deep encoder/decoder layered architecture for precise salient-object localization) and center-surround connections to exploit multi-level features.

3.2.3/ VISUAL ATTENTION FOR DRIVING ENVIRONMENT

Visual saliency detection while driving has become an important topic for research in intelligent vehicle systems. The driving environment, especially in an urban scenario, is extremely complex and the driver should pay more attention to various objects and regions while driving. The visual saliency detected/predicted by the saliency model may not be viable for the real driving scene. There is a lack of experimental research in this area, as well as a lack of saliency datasets for driving.

Currently, visual attention models for the driving environment refer to the actual attention and gaze of the human driver, as well as fixations of the region based on eye- position cues or traffic light/sign detection. Over the years, several saliency datasets for driving have been published to improve and advance these models. Work by Deng et al. (2014)

Deng et al. (2016) exploited the top-down saliency mechanism and built a traffic saliency model that uses eye-tracking for saliency detection. They built a database of saliency maps for driving by recording the eye movements of some experienced and less experienced drivers. Later Deng et al. (2017), proposed an attention model that predicts driver fixation positions using the Random Forest learning method. John et al. (2015) developed a method that identifies regions of interest in the image containing the traffic light using generated saliency maps. Yu et al. (2019a) presents a different approach for traffic sign detection, based on visual co-saliency that integrates bottom-up and top-down visual processing in an unsupervised manner. The model in Kim et al. (2016) estimates driver attention based on facial features and head direction information. Tawari et al. (2018) proposed a fully convolutional RNN model to replicate the driver's gaze fixations in the driving scene videos. Kuang et al. (2017) presented a fast Bayes saliency-based object suggestion generator for night driving scenes. The model computes saliency maps based on prior estimation (via edge detection), feature extraction (luminance, local contrast, and vehicle taillight map), weight estimation (using the variance of the feature of each class), and Bayes rule.

Palazzi et al. (2018) proposed a multi-branched deep architecture called **DR (eye)VE** Model for predicting the attentional focus of drivers. The proposed model combines raw visual scene data, motion information about optic flow, and semantic segmentation probability to predict driver attentional focus. They created a large dataset with more than 500K frames combining egocentric views (eye-tracking information) and vehicle-centric views (roof camera information). Xia et al. (2018) presented an attention model that uses driver eye movement to predict attention while driving. They developed the method Human Weighted Sampling (HWS) that identifies frames that are more critical driving moments and weights them according to their importance during training. Another huge contribution is that they created a large dataset that contains various driving scenes including driving at night, in rain, lane changing and following, turning, braking in crowded & congested situations, etc. Recently, a traffic saliency detection model was presented byDeng et al. (2020) to predict drivers' eye fixations in driving videos. They proposed a new dataset for traffic driving videos based on eye-tracking data collected from 28 experienced drivers watching driving videos. Several researchers have proposed visual attention models that examine driver attention without using eye-tracking or gaze data. These models are based on facial feature extraction Fridman et al. (2016) and head pose estimation Borghi et al. (2017). Tawari et al. (2018) proposed a fully convolutional RNN model to replicate the driver's gaze fixations in the driving scene videos. Kuang et al. (2017) presented a fast Bayes-based object proposal generator for night driving scenes.

3.3/ Our Approach for Visual Attention

After reviewing the literature, we wanted to test some of the saliency algorithms, both classical and deep, for their applicability in visual saliency for multiple objects in driving scenes. These algorithms are based on different mechanisms and use different views of saliency. Our goal is to detect important salient objects in the road context (i.e., car, pedestrians, and traffic lights/signs) that should receive more attention than other objects in the driving scene. **Figure 3.2** provides the result of the tested algorithms (GBVS Harel et al. (2007), Itti Itti et al. (1998), SR Hou and Zhang (2007), SignatureSal Hou et al. (2011), ML -Net Cornia et al. (2016), BMS Zhang and Sclaroff (2013), iSEEL Tavakoli et al. (2017), VSF Montabone and Soto (2010)). All tested algorithms resulted in different saliency maps and cannot estimate the actual saliency we aim for by considering only objects in the road context.

We propose a new visual attention framework that can detect road context objects as salient and neglect other objects in a driving scene. We focus on exploring the advantages of using conditional generative adversarial network (cGAN) in our visual attention framework to generate the saliency maps from the real scene images. **Figure 3.3** illustrates the schematic overview of the proposed visual attention framework (training and testing phases) in this work. First, we train GAN on a set of image pairs (input, target), where the input is an image from the real driving scene, while the target image is a saliency heatmap (built with the VSF Montabone and Soto (2010) saliency algorithm) of the same scene, highlighting the most salient objects as salient'. We then used the trained GAN to generate target heat- maps of unseen images. Subsections 3.3.1 and 3.4 provide details of the used GAN model Isola et al. (2017) used and the constructed heatmap dataset used for training and evaluation.

3.3.1/ Generative Adversarial Network

GAN is originally proposed by Goodfellow et al. (2014b). It consists of two competing convolutional neural networks: a generator (G) and a discriminator (D). The generator tries to generate random synthetic outputs (new data similar to the expected ones), while the discriminator tries to recognize if an input data is real (belongs to the original dataset) or fake (generated). GAN can generate good quality images from a random vector similar to the real ones. Conditional GAN (cGAN) is one of the most important extensions of the original GAN, proposed by Mirza and Osindero (2014). They add a parameter to the generator as a label that allows to condition the data generation process.

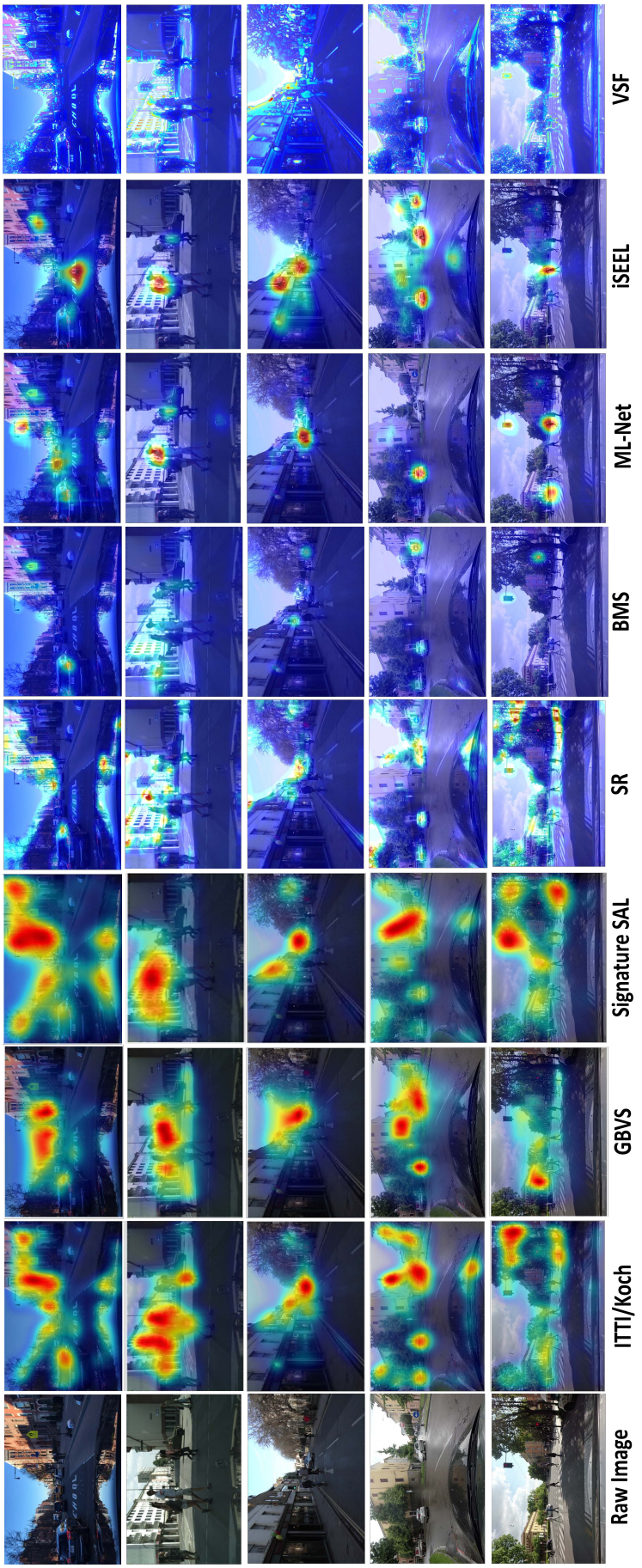


Figure 3.2: Comparison of the different saliency algorithms results

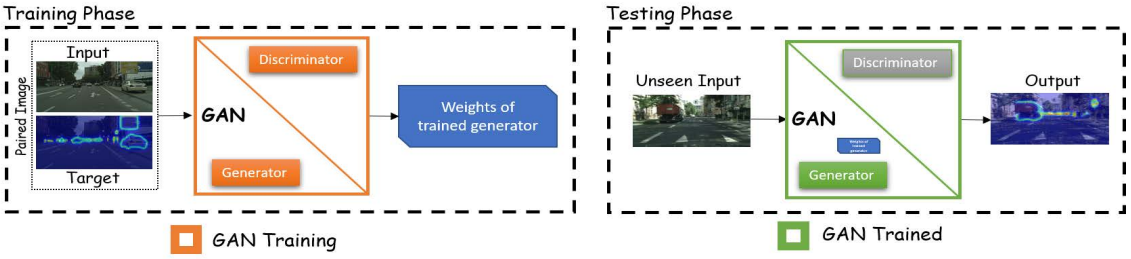


Figure 3.3: Framework - Training and Testing phases

The motivation for using a generative adversarial network is its unsupervised representation learning (e.g., it can learn from completely imaginary data). Current SOTA models for visual attention use a Convolutional Neural Network (CNN), a deep network for learning high-level, multi-scale features. These models use a binary cross-entropy loss in training, which leads to an independent prediction of the saliency probability of each pixel. This creates the problem of spatial discontinuity, and also fails to produce a fine-grained delineation of the predicted saliency maps. Over the years, several solutions have been proposed to overcome these problems by using superpixel segmentation, Conditional Random Field (CRF) as post-processing, etc. All these approaches are complex and time consuming. Using GAN for our framework has an advantage over a pixel classification based CNN network because we only use the GAN generator part, which is a simple encoder/decoder architecture with few layers. Moreover, training a pixel classification based CNN network requires labeling every pixel in the image, which is time and labor intensive. Therefore, GANs can be trained easily if a well-paired dataset is provided and good synchronization between the generator and discriminator gives good results.

We borrow the **pix2pix** Isola et al. (2017) GAN architecture, which is suitable for image-to-image translation tasks and can be conditioned on the input image to generate the corresponding output image. **Figure 3.4** shows the structure of the used GAN pix2pix. The generator network is based on **U-Net** Ronneberger et al. (2015) architecture, modified by introducing multiple skip connections between layers. The architecture consists of an encoder network that extracts the image features of the input images and a decoder network that recovers the image features and increases the image resolution using the output of the encoder. Skip connections are employed to concatenate all the channels at layers from the encoder to the decoder for improving mapping performance. The discriminator network uses patch-based assessment, a **PatchGAN** classifier that classifies each $N \times N$ patch in the input image as real or fake by convolution. Such a network structure takes advantage of fewer network parameters for training and gives good results in discriminating between real and fake images. An illustration of the U-network and PatchGAN is given in **Figures 3.5** and **3.6**, respectively. In the network model, both generator and discriminator use modules of the 2D convolutions, batch normalization, dropout (generator only) and activation layers.

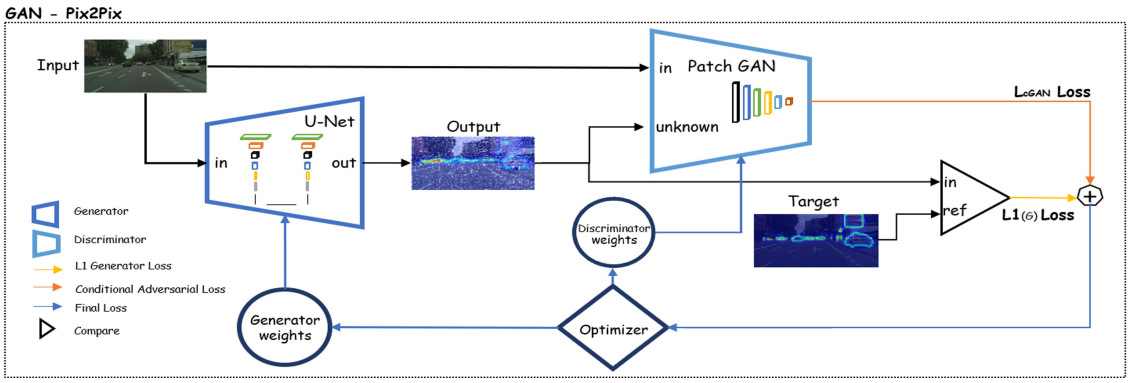


Figure 3.4: GAN Architecture used in our framework

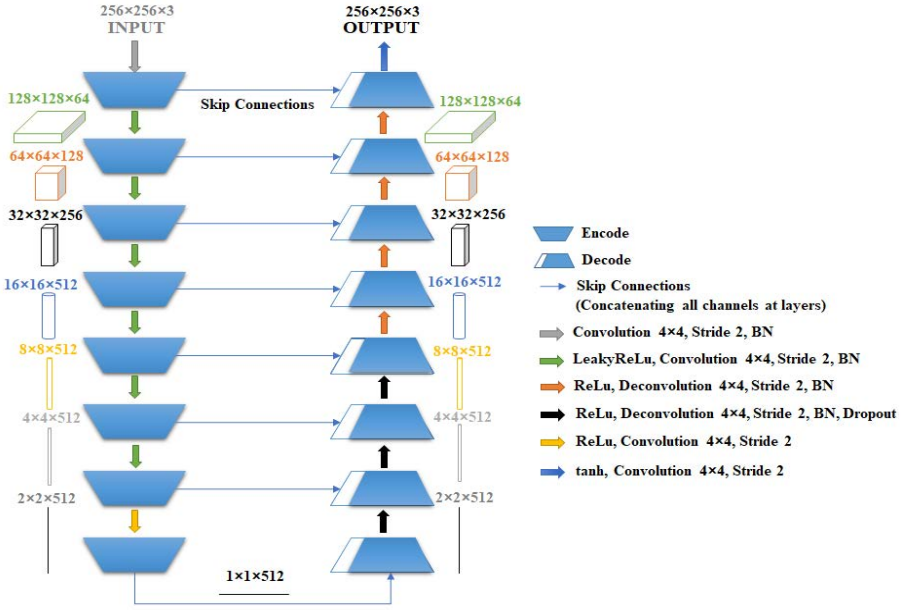


Figure 3.5: U-Net structure encoder/decoder Network

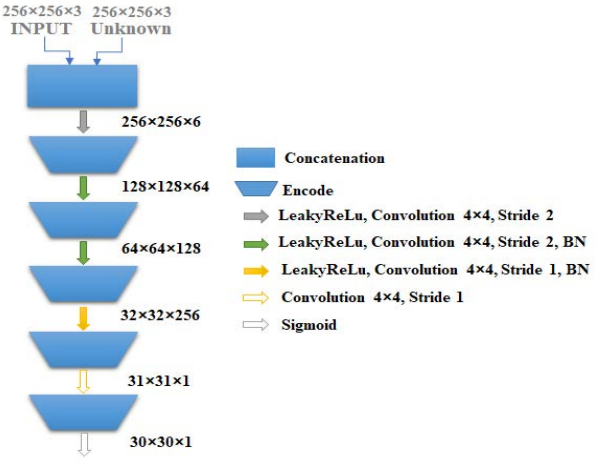


Figure 3.6: PatchGAN Network

Table 3.1: **Notation Overview**

Term	Meaning	Distribution	Meaning
x	Real Image	p_{data}	Real and target training data distribution
y	Target Image	p_z	Noise distribution (*e.g. N(0; 1)*), $z \sim p_z$
g	Generated Target Image	p_y	Known target distribution, $y \sim p_y$
z	Noise	p_x	Real data distribution, $x \sim p_x$
μ	Average	p_g	Generated Target data distribution

The objective function is summarized as follows:

$$\begin{aligned}\mathcal{L}_{cGAN}(G,D) = \mathbf{E}_{x,y}[logD(x,y)]+ \\ \mathbf{E}_{z,x}[log(1-D(G(z,x),x))]\end{aligned} \tag{3.1}$$

The generator tries to minimize $log(1-D(G(z,x),x))$ while discriminator tries to maximize $logD(x,y)$, following the min-max optimization rule:

$$\begin{aligned}\min_G \max_D \mathbf{E}_{x,y\sim p_{data(x,y)}}[logD(x,y)] \\ +\mathbf{E}_{z\sim p_z, x\sim p_x}[log(1-D(G(z,x),x))]\end{aligned} \tag{3.2}$$

thus

$$G^* = arg\min_G \max_D \mathcal{L}_{cGAN}(G,D)$$

The L1 loss Bloomfield and Steiger (1983) ($\mathcal{L}_{L1}$) is combined with the conditional adversarial loss($\mathcal{L}_{cGAN}$) which encourages less blurring:

$$\mathcal{L}_{L1}(G) = \mathbf{E}_{x,y,z}[\|y - G(z,x)\|_1] \tag{3.3}$$

The final objective is then:

$$G^* = arg\min_G \max_D \mathcal{L}_{cGAN}(G,D) + \lambda\mathcal{L}_{L1}(G) \tag{3.4}$$

where λ is a regularization constant which is set to 100 as reported in Isola et al. (2017). **Table 3.1** shows the summary of the notations.

3.4/ PROPOSED BENCHMARK

Data is considered as the backbone for developing machine learning systems. Nowadays the data for saliency model development is collected from the fixation or gaze of the human eye. This data as a saliency map (gray scale or heat map image) is obtained using a Gaussian probability function, which indicates the probability of each image pixel attracting human attention. Several saliency-based benchmarks have been created using eye-tracking data or by observing human behavior while driving. The Berkeley DeepDrive

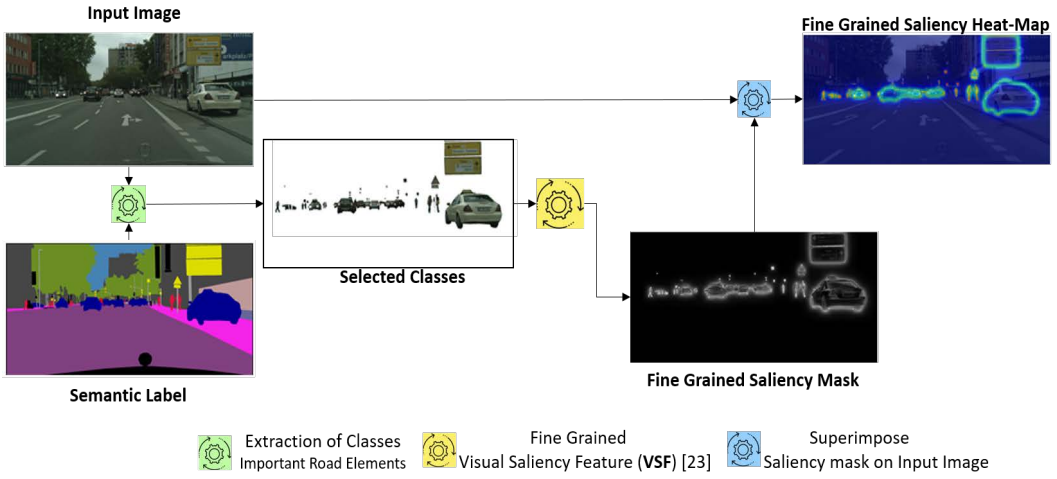


Figure 3.7: Data gathering through different processes

Laboratory developed a large-scale driving dataset called Berkeley DeepDrive Attention Xia et al. (2018), consisting of 1232 videos containing attention in critical situations. They used an eye movement averaging technique (averaging the gaze of multiple human observers) to remove the unimportant objects such as buildings, vegetation, trees, poles, etc. Alletto et al. (2016) presented a large dataset Dr (eye)VE with 500,000 frames and 6 hours of driving data at different times of the day, weather and traffic conditions. However, the attention maps are collected and ranked on one driver's perspective. Other datasets Fang et al. (2019) Underwood et al. (2011) Simon et al. (2009) are also based on gaze information from fixations. Few researchers use mouse click method or webcams for data collection to reduce time and labor cost, but it lacks accuracy. We propose a different approach for data acquisition by using the semantic label information of driving scene datasets. **Figure 3.7** shows the different processes performed for data collection (saliency heat map generation).

3.4.1/ Object/Class Selection

The objects in the driving scene can be ranked or prioritized according to their importance or relevance to safe driving. Depending on the driving situation, human drivers make decisions and prioritize more relevant objects over less relevant ones (e.g., people over animals, pedestrians over cars), but how would a machine make such decisions in advance? A good article Awad et al. (2018) from MIT probes public opinion on this question. Several things affect driving situations, such as each road user and object in the scene, the driver's state and experience, and also the vehicle being driven. According to the somatic marker hypothesis Fuller (2011), the attention priority given to objects in the driving scene is a function of the strength of the driver's sense of risk. The objects that receive higher ratings of sense of risk and attention are vehicles, pedestrians, traffic

Table 3.2: **Summary of Datasets**

Dataset	Samples	
	Training	*Validation*
Berkeley Deep Drive Yu et al. (2018)	7000	1000
CamVid Fauqueur et al. (2007)	367	101
Cityscapes Cordts et al. (2016)	2975	500
VADD	10342	1601

lights, and traffic signs. Similar risk sense responses are obtained by tracking the sequence of the driver's eye fixations on road objects while viewing the driving scene. Our object class selection for saliency heatmap data generation is also based on these road objects, i.e. persons (pedestrians, cyclists), vehicles (cars, motorcycles, trucks, trams), and others (traffic lights/traffic signs). The attending driving-specific salient features are to be used as input for decision making and/or planning or monitoring. We incorporate three driving datasets BDD, Cityscape, and CamVid (**Table 3.2**) that provide semantic labels (annotation of each object in images).

3.4.2/ Saliency Algorithm Selection

Numerous saliency works have models on various metrics, noise robustness, and side-by-side comparison of computed saliency maps (visualization) Bylinskii et al. (2018) Kim and Milanfar (2013) compared. We study the robustness to noise of saliency algorithms, Itti, GBVS, SR, ML -Net, BMS, iSEEL, and VSF (presented in subsections 3.2.1 and 3.2.2). Our goal is to choose the better saliency detection algorithm for constructing ground truth for our desired application of visual attention. The white Gaussian noise is added to 500 test images with a mean of zero and three different variance values σ^2 (0.04, 0.12 and 0.19) as shown in **Figure 3.8**. We fed clean and noisy images into the saliency detection algorithms and used the matrices Peak Signal to Noise Ratio (PSNR) and Mean Squared Error (MSE) (3.5.3) for evaluation. The (**VSF**) Montabone and Soto (2010) algorithm shows more stable results (with low MSE and high PSNR) as shown in **Table 3.3**, and provides the complete shape of the highlighted objects.

Table 3.3: **Noise robustness based saliency algorithm evaluation**

Saliency Methods	MSE (Low is good)			PSNR (High is good)		
	σ^2=0.04	σ^2=0.12	σ^2=0.19	σ^2=0.04	σ^2=0.12	σ^2=0.19
Itti/Koch	1127.56	1109.98	1109.98	18.0269	18.0985	18.1531
BMS	1228.03	1399.94	1399.94	18.1355	17.3990	17.1982
ML-Net	693.624	755.848	755.848	19.9791	19.6315	18.8982
SR	1366.642	2306.73	2306.73	19.6558	16.9105	14.7811
GBVS	1153.76	1338.19	1338.19	18.2219	17.5340	17.2169
iSEEL	1011.23	1209.01	1209.01	19.7487	18.6684	17.9921
VSF	**55.5966**	**180.129**	**180.129**	**31.4493**	**25.7475**	**23.3944**

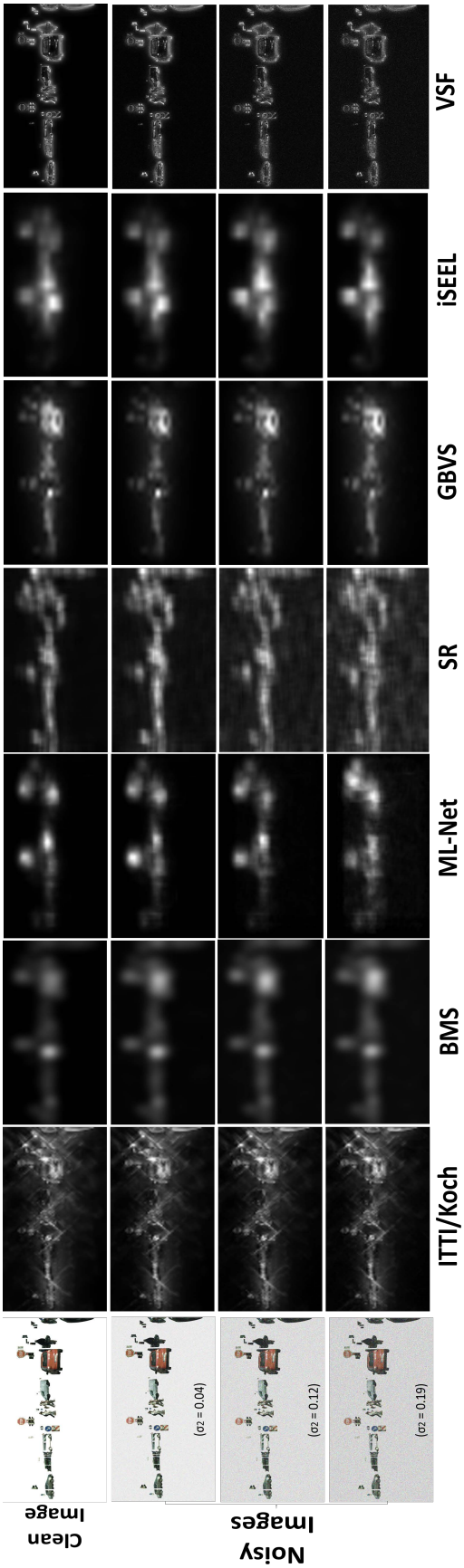


Figure 3.8: The results of the saliency algorithms given a noisy image. The added noise is a white Gaussian noise with different variance σ^2 values

The shape property is so important in our application that the driver can easily recognize any highlighted object in the scene when we integrate this framework into a Advanced Driver Assistance System (ADAS) or 3D driving simulator. Moreover, the full object shape is useful for semantic segmentation because the computer can quickly process the object shapes from the heat map to segment the important classes.

Finally, we created heatmaps from the grayscale masks and overlaid them on the original images to obtain a saliency heatmap that highlights the selected class objects as the most prominent and salient regions in the images.

3.5/ Experimental Analysis

We first describe the configurations of the model GAN (training/ Testing Protocols). Then we present the data used for training & testing. Next, we present the metrics used for performance evaluation.

3.5.1/ Model Configuration

Training/Testing Protocols

As previously defined, the generator of GAN is an encoder-decoder architecture using a U-network, and the discriminator design is based on the PatchGAN model. The generator network consists of 2D convolutional blocks, batch normalization, dropout and activation layers. In the last layer of the generator, the activation function tanh is used (produces image pixel values in the range [-1,1]). In the discriminator model, we tested discriminator with two different patch sizes, 70×70 PatchGAN and 1×1 PixelGAN. These models take two concatenated images as input and classify whether the patch output is real or fake. The discriminator model is trained with real and generated images, and the generator model is trained by the discriminator model. The generator is updated to minimize the *L1* loss between the target and generated images. The discriminator uses a sigmoid function in the last layer. The model is optimized with binary cross entropy, and the momentum is set to 0.5. The batch size is set to 1. The learning rate is initially set to 0.0002 and linearly decaying close to zero after 150 epochs. The learning process stops after 300 epochs. To reduce the training time, the images are resized to 512×512. The model is trained using an NVIDIA GTX 1080 Ti 12GB GPU, and the GAN model implementation is based on PyTorch.

3.5.2/ BENCHMARKS

The framework is evaluated with three driving datasets, Berkeley Deep Drive (BDD), Cityscapes and CamVid. We used a cross-validation protocol that resulted in 3 training sessions and 9 evaluation experiments. The datasets used for each training are: 7000 for $Train_{BDD}$, 2975 for $Train_{Cityscapes}$, and 367 for $Train_{CamVid}$. For validation, we considered 1000, 2975, and 367 number of images for Val_{BDD}, $Val_{Cityscapes}$, and Val_{CamVid}, respectively. We also trained the model with data combining all three datasets (10342 images), and evaluated each dataset.

3.5.3/ EVALUATION METRICS

We evaluated our results using several quantitative metrics. Saliency algorithms are evaluated using MSE and PSNR.

1) **Mean Squared Error** (MSE), representing average of the squares of the errors between clean image and degraded noisy image.

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}\|a(i,j) - b(i,j)\|^2 \tag{3.5}$$

where *a* is the matrix data of the original clean image, *b* is the matrix data of the degraded noisy image. *m* represents the number of rows and *n* serves as the number of columns of the images. ***i*** and ***j*** are indexes for these rows and columns.

2) **Peak Signal to Noise Ratio** (PSNR) is a ratio between the maximum and minimum possible values of a changeable quantity.

$$PSNR = 20log_{10}(\frac{MAX_a}{\sqrt{MSE}}) \tag{3.6}$$

where MAX_a is the maximum value that exists in original clean image.

The results of our framework are evaluated using the SSIM, FID and WD metrics. For saliency map evaluation (comparison with SOTA methods), we used four evaluation metrics that are commonly used for saliency attention map prediction: Kullback-Leibler divergence (KL -Div), Correlation Coefficient (CC), Area under Curve - Judd (AUC-Judd) and Normalized Scanpath Saliency (NSS).

1) **Structure Similarity Index** (SSIM) Wang et al. (2004), is a widely used metric that measures the structural or perceptual difference between two images. SSIM includes important structural information (luminance and contrast), which means that nearby pixels have strong dependencies on each other and carry information about the structure of objects in the visual scene. Luminance tends to be less visible in bright regions, while

contrast tends to be less visible where there is significant activity in the image. SSIM ranges from ***0*** to ***1***, the higher the better.

The SSIM metric is calculated on multiple windows of an image. The SSIM is expressed as follow:

$$\mathcal{SSIM}(d, \hat{d}) = \frac{(2\mu_d\mu_{\hat{d}} + c_1)(2\sigma_{d\hat{d}} + c_2)}{(\mu_d^2 + \mu_{\hat{d}}^2 + c_1)(\sigma_d^2 + \sigma_{\hat{d}}^2 + c_2)} \tag{3.7}$$

where σ is variance, $\sigma_{d\hat{d}}$ is covariance, c_1 and c_2 are two variables used to stabilize the division. $c_1 = (k_1 L)^2$ and $c_2 = (k_2 L)^2$. L is the dynamic range of the pixel-values (i.e. $2^{Bits/Pixel} - 1$) and k_1= 0.01 and k_2 = 0.03 by default.

2) **Frechet Inception Distance** (FID) Heusel et al. (2017), is a well-known metric used to evaluate GANs. FID is an improved version of Inception Score Salimans et al. (2016), which uses a pre-trained inception model Szegedy et al. (2016) (trained on ImageNet) to measure the objectiveness and diversity of generated images. FID compares the statistics of real target and generated target samples using the Frechet distance between two multivariate Gaussians.

$$InceptionScore = \mathbf{E}_{g\sim P_g} D_{KL}(p(y|g)||p(y)) \tag{3.8}$$

Equation 3.8 compares the real target distribution $(p(y|g)$ low entropy with the generated target distribution $p(g) = \int_g p(y|g)p_g(g)$ high entropy, and KL-divergence between them.

$$FID = ||\mu_d - \mu_{\hat{d}}||_2^2 + Tr(\Sigma_d + \Sigma_{\hat{d}} - 2(\Sigma_d\Sigma_{\hat{d}})^{\frac{1}{2}}) \tag{3.9}$$

where *Tr* refers to trace of matrix. Lower FID score indicates less diversity between real and generated images.

3) **Wasserstein distance (WD)** Huang et al. (2018a) is the measure of distance between two probability distributions P_d and $P_{\hat{d}}$.

$$WD(P_d, P_{\hat{d}}) = \inf_{\gamma\in\Gamma(P_d,P_{\hat{d}})} \mathbf{E}_{(s^d,s^{\hat{d}})\sim\gamma}[D(s^d, s^{\hat{d}})] \tag{3.10}$$

where $\Gamma(P_d, P_{\hat{d}})$ denotes the set of all joint distributions (i.e. probabilistic couplings), and D $(s^d,\ s^{\hat{d}})$ denotes the base distance between the two sample images. The smaller the Wasserstein distance, the more similar two distributions are.

4) **Kullback-Leibler divergence (KL-Div)** Bylinskii et al. (2018) is an asymmetric dissimilarity metric, which measures the difference between two probability distributions P_d and $P_{\hat{d}}$.

$$KL(P_d, P_{\hat{d}}) = \sum_i P_{\hat{d}_i} log(\epsilon + \frac{P_{\hat{d}_i}}{\epsilon + P_{d_i}}) \tag{3.11}$$

where ϵ is a regularization constant. Lower the KL score, better the approximation.

5) **Area Under Curve (AUC)** proposed by Judd Bylinskii et al. (2016), measures the trade off between true and false positives distinguished by different thresholds using the saliency map as a binary classifier. The true positives are saliency map values above the threshold of fixed pixels, and their ratio to the total number of fixations is called the true positive rate (T_P rate). The false positives are saliency map values above the threshold at non-fixed pixels, and their ratio to the total number of saliency map pixels at a given threshold is called the false positive rate (F_P-rate) Bylinskii et al. (2018).

6) **Linear Correlation Coefficient(CC)** Bylinskii et al. (2018), is a measure of the linear relationship between saliency map (P_d) and fixation map ($P_{\hat{d}}$).

$$CC(P_d, P_{\hat{d}}) = \frac{\sigma(P_d, P_{\hat{d}})}{\sigma(P_d) \times \sigma(P_{\hat{d}})} \tag{3.12}$$

where $\sigma(P_d, P_{\hat{d}})$ is the covariance of (P_d) and $(P_{\hat{d}})$.

7) **Normalized Scanpath Saliency (NSS)** Bylinskii et al. (2018), is measured by taking the average of the values in a saliency map (P_d) normalized to have a mean of zero and a standard deviation of one unit at a binary map of fixation locations ($P_{\hat{d}}$).

$$NSS(P_d, P_{\hat{d}}) = \frac{1}{N} \sum_i \overline{P}_{d_i} \times P_{\hat{d}_i} \tag{3.13}$$

where i indexes the i^{th} pixel, and N is the total number of fixated pixels; $N = \sum_i P_{\hat{d}_i}$ and $\overline{P}_d = \frac{P_d - \mu(P_d)}{\sigma(P_d)}$

3.5.4/ Results and Discussion

This section is divided into two subsections. In the first subsection, we summarize the quantitative and qualitative performance of our proposed framework. The second subsection presents the comparison of the proposed framework with the SOTA saliency and eye fixation network models.

3.5.4.1/ Proposed Framework Experiments

We trained the model on each of the three datasets and performed cross-validation to compare performance (scores in black). **Table 3.4** shows the quantitative performance

Table 3.4: **Quantitative Performance**

	Database	$No_{of Images}$	Validation											
			BDD			CamVid			Cityscapes			VADD		
			1000			101			500			1601		
			SSIM 0 - 1	WD 0 - ∞	FID 0 - ∞	SSIM 0 - 1	WD 0 - ∞	FID 0 - ∞	SSIM 0 - 1	WD 0 - ∞	FID 0 - ∞	SSIM 0 - 1	WD 0 - ∞	FID 0 - ∞
Train	BDD	7000	0.8016	1.579	22.03	0.7945	2.96	93.99	0.7825	1.803	43.62	0.7941	3.12	19.90
	CamVid	367	0.0597	16.01	54.88	0.7967	2.84	83.81	0.6742	10.08	70.74	0.4634	9.35	40.83
	Cityscapes	2975	0.7211	2.083	28.54	0.7891	4.65	102.03	**0.8115**	**1.60**	**40.97**	0.7704	3.55	21.87
	VADD	10342	**0.8064**	**1.293**	**20.97**	**0.8286**	**2.61**	**74.94**	0.8042	1.75	41.04	**0.8022**	**2.65**	**18.77**

on the validation datasets. We also trained the model on the combined dataset (which combines all three datasets and is called VADD) and evaluated performance (scores in blue). The model trained with the combined dataset scored low on the cityscapes validation set compared to the model trained with just cityscapes. This fact is due to that BDD constitutes around 2/3 of the combined dataset and the conditions that occur in BDD are larger than those in cityscapes (e.g., night, snow, rain, etc.). Therefore, the additional conditions may affect the GAN training ("over tuned"). The results show that the model performs better when trained with the combined training dataset. In the metrics, the SSIM score ranges from 0 - 1, meaning that closer to 1 is better, and for WD / FID, the lower the better, ranges from 0 - ∞.

Figure 3.9 shows the visual results of our framework on the validation set (2 samples from each subset in rows 1 2 and 3). Several objects such as vehicles ahead, cyclists, traffic lights/signs and pedestrians nearby require attention consistently, and it can be seen that our framework accurately highlights these objects as salient, similar to the targets (ground-truth). In **Figure 3.10**, we tested our framework on images from the validation set with different environments and adverse driving conditions, such as rain, fog, snow, night, city traffic, highways, bridges, and tunnels. We evaluated the performance, as shown in **Table 3.5**. The model scores slightly low on night images, which is due to the smaller number of night images in the dataset compared to the other images.

Figure 3.9: Visual results on VADD validation set

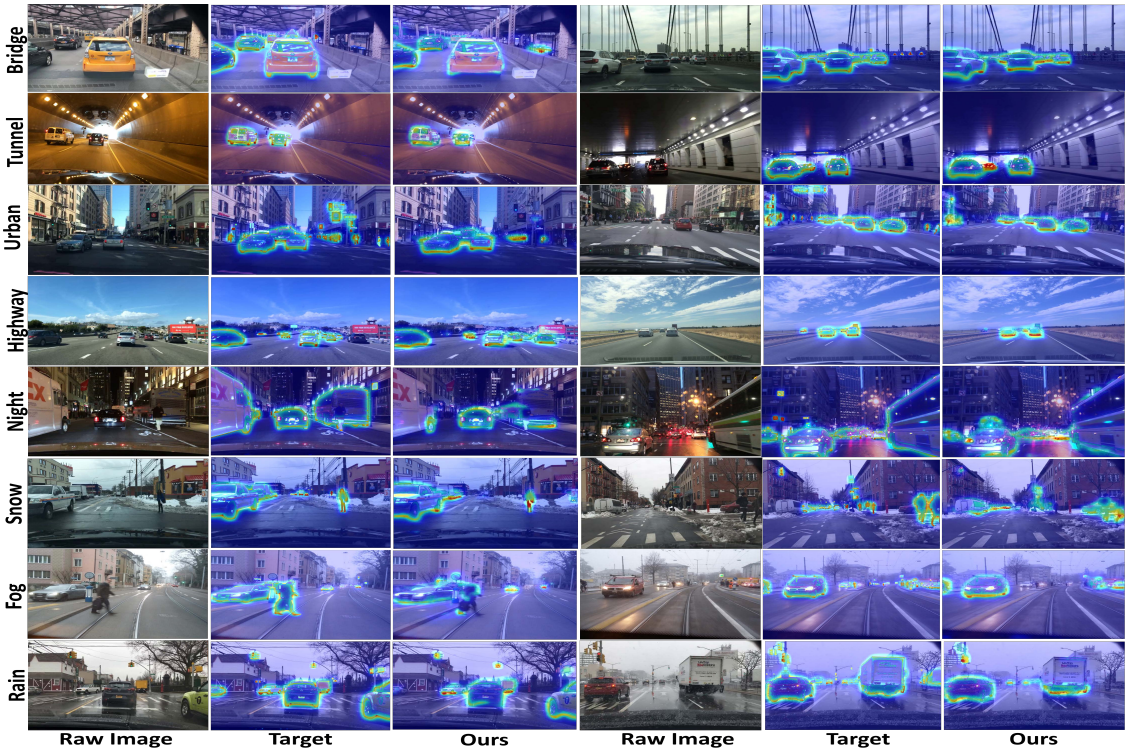


Figure 3.10: Visual results on different environments and weather conditions

Table 3.5: **Quantitative Performance** in Different Environment Conditions

	Diverse Conditions	SSIM 0 - 1	WD 0 - ∞	FID 0 - ∞
Weather	Rain	0.7711	1.98	40.79
	Fog	0.7601	2.08	59.80
	Snow	0.8084	1.70	31.14
Environment	Tunnel	0.8051	1.90	39.58
	Night	0.5725	9.91	101.12
	Bridge	0.8225	1.62	30.47
	Highway	0.7588	2.58	82.22
	Urban	0.8567	0.97	22.61

The overall performance of the framework is really good compared to the targets as it detects or pays attention to important objects like vehicles, pedestrians and traffic lights while ignoring irrelevant objects like buildings, trees etc.

The performance of deep learning models decreases when they are evaluated on datasets that were not used for training. We wanted to test our framework on datasets other than those used in training. We considered two datasets; first, the EU long-term dataset developed by Yan et al. (2019), and second, the Synthia dataset developed by Ros et al. (2016b). The EU long term dataset is a public dataset for autonomous driving covering different environments, seasons, weather and lighting conditions. Synthia is an extensive public dataset with synthetic images for driving scenes. From **Figure 3.11**, we can see the promising results of our framework. We also tested our model on unseen

random images from the Internet and obtained good results, as shown in **Figure 3.12**.

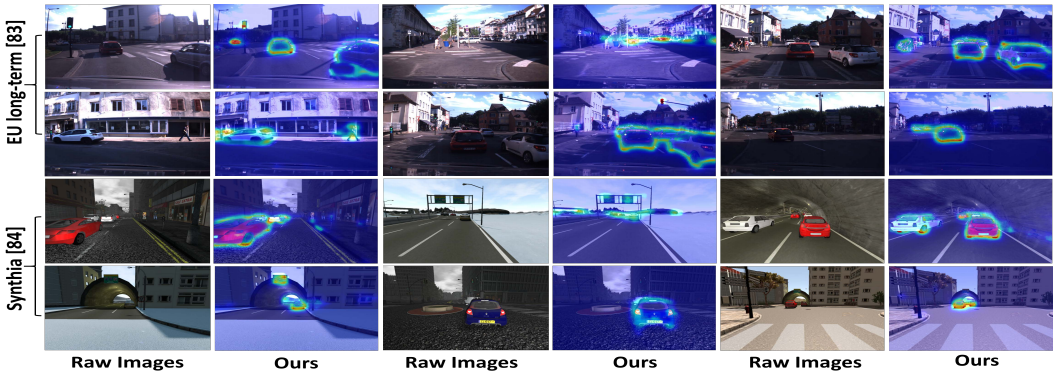


Figure 3.11: Test on EU long-term and Synthia Datasets

Figure 3.12: Random unseen images

3.5.4.2/ SOTA Comparison

We proposed a new idea of data generation and framework for visual attention prediction, and we believe that it is necessary to compare our visual attention performance both quantitatively and qualitatively with SOTA saliency network models and eye fixation attention models.

AGAINST SALIENCY NETWORK MODELS

The first comparison experiment is performed against the state-of-the-art methods ML -Net Cornia et al. (2016) and SAM -Net Cornia et al. (2018). We used the codes provided by the authors, trained these networks on our proposed VADD training set of 10342 images, and ran tests on a validation set of 1601 images. The experiment was implemented using Python 3.5 and PyTorch on an NVIDIA 1080Ti GPU. During the training process, the Adam optimizer was used with an initial learning rate of 0.0005 and an MSE loss function. The experiment was trained with 500 update epochs. Four different metrics (KL -Div, AUC-Judd, NSS and CC) were used to evaluate network models, **Table 3.6** documents the obtained values. The proposed framework outperforms the state-of-the-art saliency networks on the VADD dataset in terms of all evaluation metrics. **Figure 3.13** illustrates the results of the evaluated methods. The proposed method captures better results (closer to ground truth) compared to the other saliency network models. For example, the outlines of salient objects are very clear, especially when the objects are distant or small.

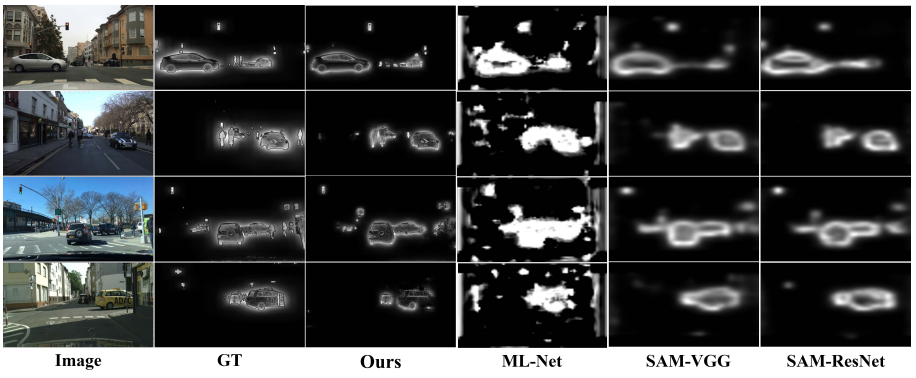


Figure 3.13: Comparison of the proposed framework with the saliency networks (ML-Net, SAM-Vgg and SAM-ResNet) on VADD validation set. It can be seen that, our proposed framework captures better results, more detailed and closer to the ground truth (GT) targets.

Table 3.6: **Quantitative Performance** Vs **Saliency Network Models** on VADD validation set

Network Models	**KL-Div** ↓	**AUC** ↑	**NSS** ↑	**CC** ↑
ML-Net Cornia et al. (2016)	2.737	0.595	1.690	0.620
SAM-VGG Cornia et al. (2018)	1.915	0.691	1.703	0.685
SAM-ResNet Cornia et al. (2018)	1.757	0.718	1.720	0.697
Proposed	**1.450**	**0.754**	**1.969**	**0.736**

In the second experiment, we trained and tested the proposed framework on the well-known SALICON dataset Jiang et al. (2015). We compared our results both quantitatively

and qualitatively with SOTA methods. We calculated the mean prediction errors by applying three different metrics (AUC-Judd, NSS, and CC) on the SALICON test set. The results are shown in Table 3.7 where the scores of the compared methods are taken from the original papers. The first three methods are traditional methods and the last three are deep learning based models. As can be seen from Table 3.7, the proposed framework achieves high scores in the AUC and NSS metrics compared to the other methods. For CC metric, the proposed method provided the second better score, with 0.003 difference from SAM -ResNet, which had the best score. **Figure 3.14** shows a qualitative comparison of the methods on the SALICON test dataset. The predicted saliency maps of our proposed framework are much closer to the ground truth fixation maps compared to the others.

Table 3.7: **Quantitative Performance** Vs **Saliency Network Models** on SALICON test Dataset

Network Models	AUC ↑	NSS ↑	CC ↑
Itti Itti et al. (1998)	0.667	-	0.205
GBVS Harel et al. (2007)	0.789	-	0.421
BMS Zhang and Sclaroff (2013)	0.789	-	0.427
ML-Net Cornia et al. (2016)	0.866	2.789	0.743
SAM-VGG Cornia et al. (2018)	0.881	3.143	0.825
SAM-ResNet Cornia et al. (2018)	0.883	3.204	**0.842**
Proposed	**0.889**	**3.231**	0.839

Against Eye Fixation Attention Models

Two experiments were conducted to compare the results of the proposed framework with visual attention models that use eye fixation data for prediction in driving scenes: **1)** Train and test the SOTA visual attention model BDDA Xia et al. (2018) directly on the proposed VADD dataset. **2)** Train the proposed framework on the BDDA dataset and perform comparison with SOTA methods.

Comparison on VADD:

We trained the Berkeley DeepDrive Attention (BDDA) model Xia et al. (2018) on our proposed VADD benchmark. The authors provided the code and all the details to train and test their model, we trained the model from scratch. **Table 3.8** and **Figure 3.15** present the quantitative and qualitative evaluation of the BDDA model against the proposed framework on the VADD validation set.

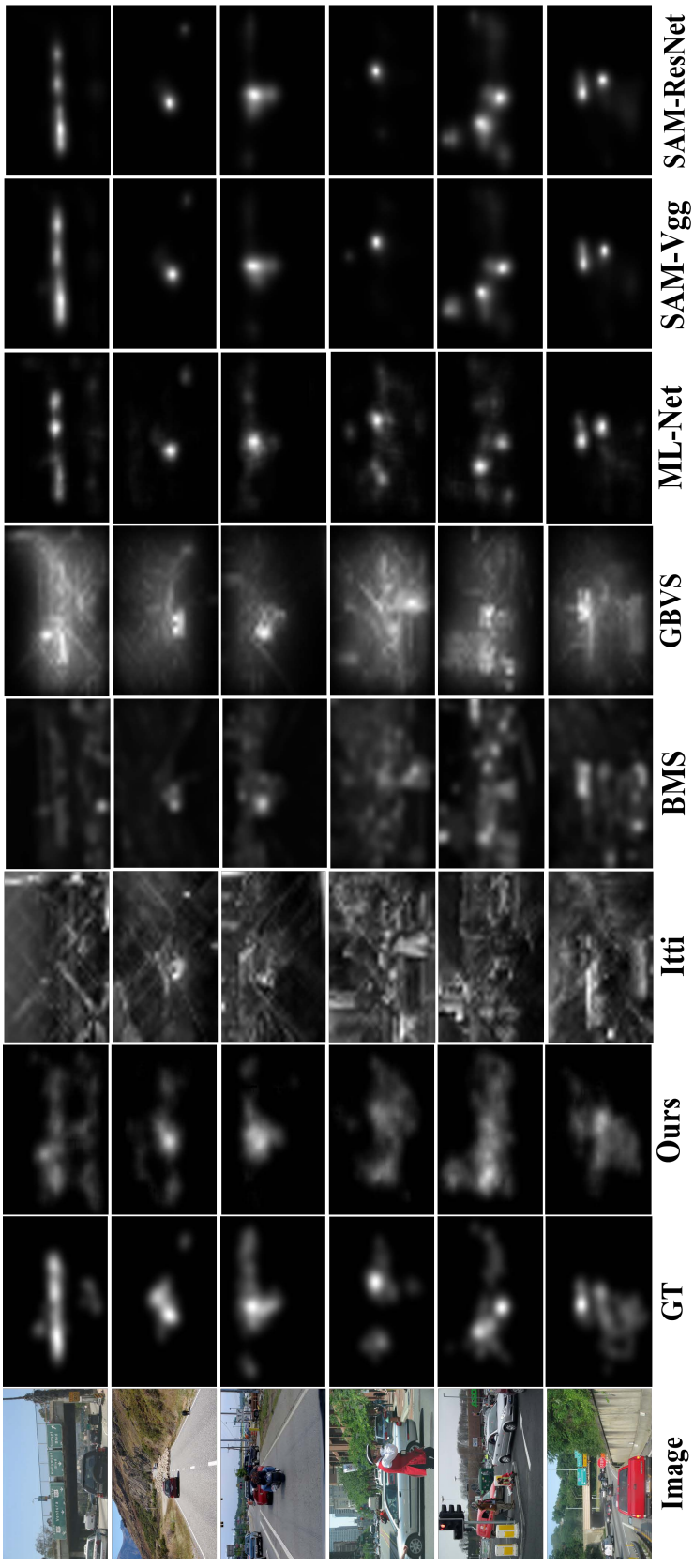


Figure 3.14: Comparison with different saliency algorithms both classic and deep leaning ones on the SALICON testing set.

Our framework outperforms the BDDA model on the VADD dataset in all evaluation metrics. In Figure 3.15, we can clearly see that the fixation-based approach BDDA is not able to clearly highlight objects (with their outlines and boundaries), moreover, multiple salient objects are connected even if they are far away from each other. We also find that the BDDA model leads to a lot of wrong predictions almost in every saliency map output. Moreover, the output saliency maps are very low resolution (80 ×60) and are scaled up to the size of the input image, which drastically reduces the accuracy of the prediction. In our framework, the saliency map is exactly the same size as the input image. Our intention in this comparison was to test how well the BDDA model performs on the proposed dataset. We also attempted to train or fine-tune other eye fixation models for driving on the VADD dataset, but were unable to prevail. The authors only provided the demonstration code and not the training source code.

Table 3.8: **Quantitative Performance** Vs **BDDA - Driving Attention Network Model** on VADD validation set

Network Models	**SSIM** 1 - 0	**WD** 0 - ∞	**FID** 0 - ∞
BDDA Xia et al. (2018)	0.7081	5.54	29.68
Proposed	**0.8022**	**2.65**	**18.77**

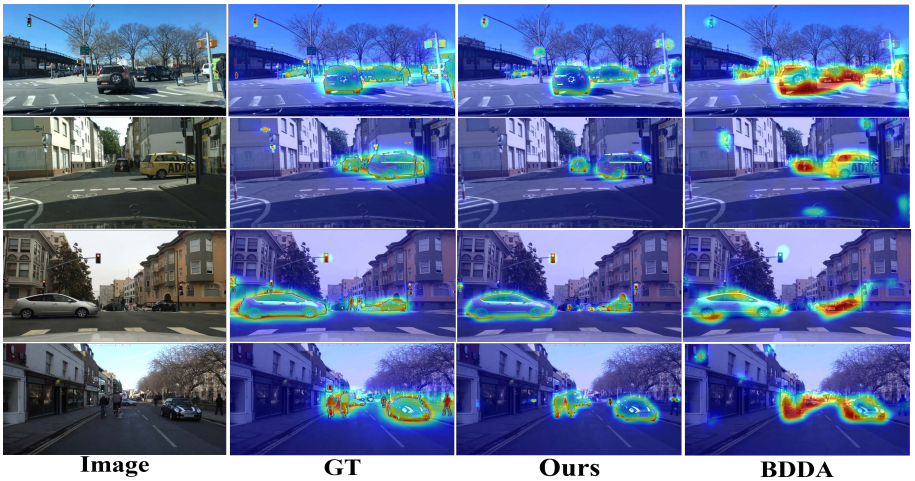


Figure 3.15: Comparison of the BDDA Network with our proposed framework on the VADD validation set. Our framework outcomes are better with clear object boundaries and outlines (close to GT) compared to the BDDA model.

Comparison on BDDA:

The proposed framework was trained on the BDDA dataset and compared both qualitatively and quantitatively with SOTA visual attention methods, as shown in **Figure 3.16** and **Table 3.9**, respectively. We used the metric scores and outcome-attention maps from the original papers. In Table 3.9, we can see that the proposed framework outperforms all

other methods with the lowest KL -Div and the highest AUC, NSS and CC -values. The qualitative comparison in Figure 3.16 shows that the results of our framework are very close to the ground truths compared to the others. The aim of this comparison was to test the performance and capability of the proposed framework (generative model) trained and tested on an eye fixation based driving attention dataset.

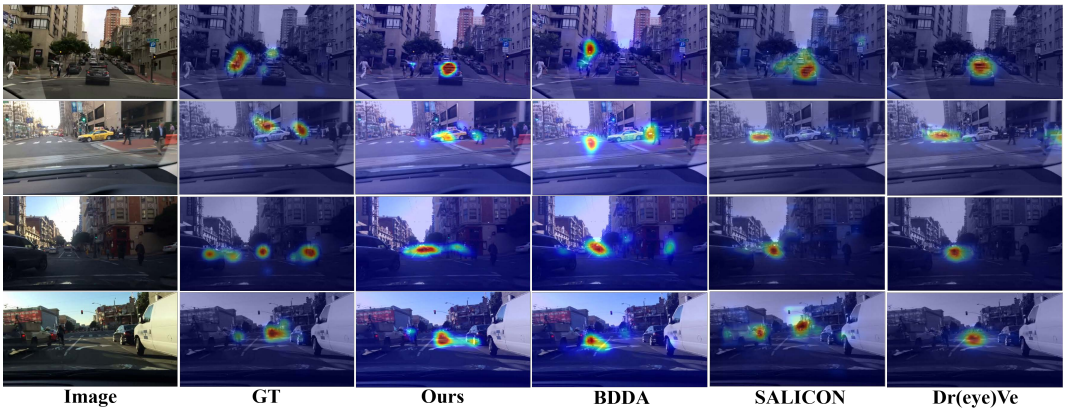


Figure 3.16: Comparison of results from our proposed framework and eye fixation attention networks on the BDDA testing Dataset

Table 3.9: **Quantitative Performance** Vs **Eye Fixation Network Models** on BDD-A testing Dataset

Network Models	KL-Div ↓	AUC ↑	NSS ↑	CC ↑
SALICON Huang et al. (2015)	1.41	0.915	3.14	0.53
Dr(eye)Ve Palazzi et al. (2018)	1.95	0.866	2.90	0.50
BDDA Xia et al. (2018)	1.24	0.931	3.51	0.59
Proposed	**1.15**	**0.947**	**3.68**	**0.60**

VADD Vs BDDA & Dr(eye)Ve Datasets:

We trained the proposed framework on the VADD dataset and compared the results visually against Dr(eye)VE's visual attention model Palazzi et al. (2018) and the Berkeley DeepDrive Attention (BDD-A) model Xia et al. (2018) on their datasets, as shown in **Figures 3.17** and **3.18**. It can be seen that the predictions of BDD-A and Dr(eye)VE models mainly focus on the middle of the road and ignore the important objects and elements in the scene. For example, in Figure 3.17, for the raw images in row 1, the Dr(eye)VE model missed traffic signs and hardly focused on the motorcyclist, and for the first image of row 2, it missed the important pedestrian. Also, the BDD-A model missed the pedestrians, traffic lights and traffic signs, as can be for the image 1 of row 1 in Figure 3.18. Compared with the results of BDD-A and Dr(eye)VE models, our proposed system successfully detected several important objects in the scenes simultaneously.

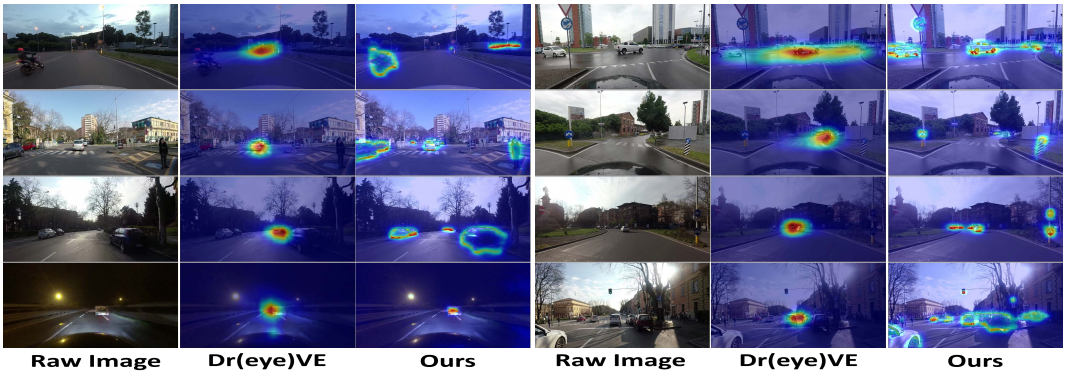


Figure 3.17: Visual Comparison with Dr(eye)VE Project results

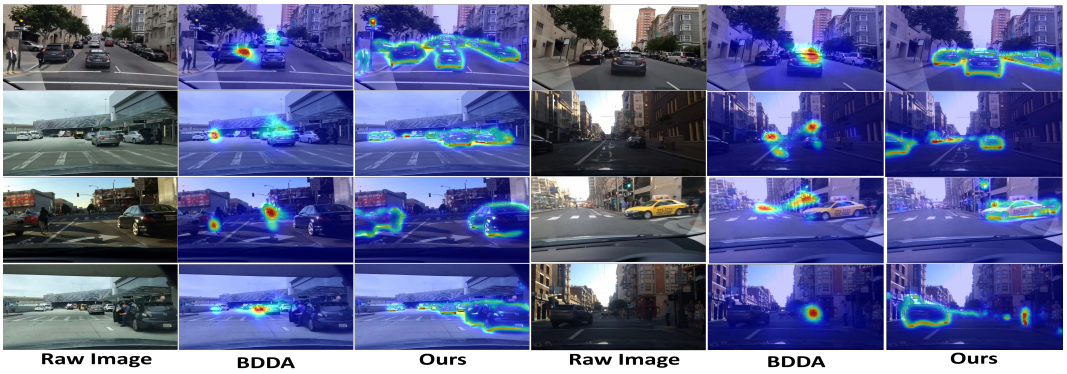


Figure 3.18: Visual Comparison with BDD-A results

3.6/ CONCLUSION

An autonomous driving system with the ability to pay attention to the most important objects/regions of the driving environment is very important to make safe driving decisions. In this chapter, we presented a new visual attention framework that highlights objects in the road context as salient based on Generative Adversarial Network. We reviewed the well-known saliency algorithms, including classical and deep learning approaches, used for visual attention. We tested these algorithms for their applicability to visual saliency for multiple objects in driving scenes. We concluded that none of these algorithms can work in complex and diverse environments such as driving. We presented a new strategy of data generation and visual saliency prediction. We investigated the noise robustness of various computational saliency algorithms on images corrupted by white Gaussian noise. The VSF algorithm was found to perform better, both quantitatively and qualitatively, for constructing data ground truth. The data are obtained from publicly available driving datasets Yu et al. (2018) Fauqueur et al. (2007)Cordts et al. (2016), which contain various driving activities and environments, including rain, night, snow, highways, and urban scenes. We evaluated our results using various metrics for quantitative performance evaluation. Experimental results, quantitative and qualitative comparisons with

SOTA saliency and eye fixation attention models demonstrated the ability of our framework to predict several important objects in interactive, complex, and dynamic driving environments.

3.6.1/ LIMITATIONS

Overall, the proposed framework performs effectively in all cases by predicting several important objects as salient in driving scenes. However, there are some limitations. First, the model predicts the target objects class-wise in the image. We assume that not all of these objects are demanding all the time, e.g., static cars parked on the roadside, distant cars, pedestrians walking on the sidewalk, some irrelevant advertising signs, etc. Second, in a few cases, the framework predict false regions as salient. For example, for raw image 1 in **Figure 3.19**, the approaching vehicle and traffic light are not detected and trees and billboards are classified as salient. Our framework also triggers false predictions due to direct sunlight or light reflections, as shown in image 2 in **Figure 3.19**.

Figure 3.19: Examples of false prediction

4

Semantic-Aware Object Identification in Urban Driving Scenarios

4.1/ Introduction

4.1.1/ Problem Statement and Motivation

In the computer vision and intelligent vehicle society, there are many interests to understand the urban driving environment by exploring the outstanding performance of artificial intelligence. Many companies and research institutions are focusing on developing intelligent vehicle systems that can automatically understand the 3D environment of the vehicle, just as humans do. Recent advances in sensor technology have led to today's vehicles being equipped with sensors that gather important information about the environment. Camera sensors, for example, provide rich color information from which semantics of the scene can be extracted. Ultrasonic sensors provide depth information for nearby hurdles. LiDAR sensors are used for accurate depth and geometry information of the environment. Many vision-based processing techniques have been developed for various purposes, such as static and moving object detection Vertens et al. (2017), depth estimation Hirschmuller (2008), traffic light detection Munoz-Organero et al. (2018), pedestrian detection Liu et al. (2019), Obstacle detection Rateke and von Wangenheim (2020), Collision warning systems Lyu et al. (2020), Lane detection Li et al. (2016b), Blind spot detection Ra et al. (2018), and so on. The image processing based techniques are relatively intuitive and less expensive than active sensing techniques. For an intelligent vehicle system, it is very crucial to thoroughly understand the status of each of the detected surrounding elements (static or moving, near or on the road), which includes object class or type, object position, direction, speed or velocity. All of this information about objects is critical because it affects the safety of the vehicle and the safety of other participants,

such as pedestrians, bicyclists, animals, and other vehicles in the scene. Moreover, such information about surrounding objects can help the system predict and plan their future state and trajectories.

We conducted a thorough literature review on the topic of semantic reasoning of the scene. We found many examples and approaches used for object detection, motion estimation, semantic segmentation, depth estimation, object tracking, and others. The current state of the art (SOTA) based image processing systems used in autonomous driving have excellent performance for the above tasks. The recent convolutional neural network (CNN) based deep learning approaches have shown amazing results in this field. The SOTA methods use implicit learning for motion information through camera sensors, LIDAR scans, inertial measurement units. However, there are still some challenges in the area of motion segmentation, i.e., successfully extracting motion information of the moving objects from a moving camera. Issues such as distortions due to motion blur, abrupt contrast changes, light reflections, and varying pixel shifts due to motion at different speeds cause problems in detecting the actual motion of the moving objects. Many methods and approaches, both classical and Deep Learning based, have been developed to solve these problems by using optical flow Yu et al. (2019b)Siam et al. (2018b)Zhou et al. (2017), Background foreground extraction Sengar and Mukhopadhyay (2020)Jung et al. (2020), LIDAR fusion Lee et al. (2020)Cho et al. (2014), Sparse feature-based methods Lenz et al. (2011), and ego-motion compensation Vertens et al. (2017). Such approaches are still inefficient and cannot effectively extract the motion information of detected moving objects from a moving camera.

In this chapter, we aim to propose a framework that combines motion and geometry related information for understanding driving environment. The ultimate goal is to extract object identification information from a moving camera, such as object class, position, motion information, and depth/distance information using image processing-based techniques. We have developed a new model for moving object detection **MOD** by integrating an encoder-decoder network with a segmentation model, and propose to use image registration as a tool for ego- motion compensation. We incorporate previously developed work on object segmentation, image registration, optical flow, and disparity estimation which could be combined with the proposed MOD to achieve our above aim. **Figure 4.1**, shows specific blocks that highlight the main steps of our framework for object identification (**FOI**).

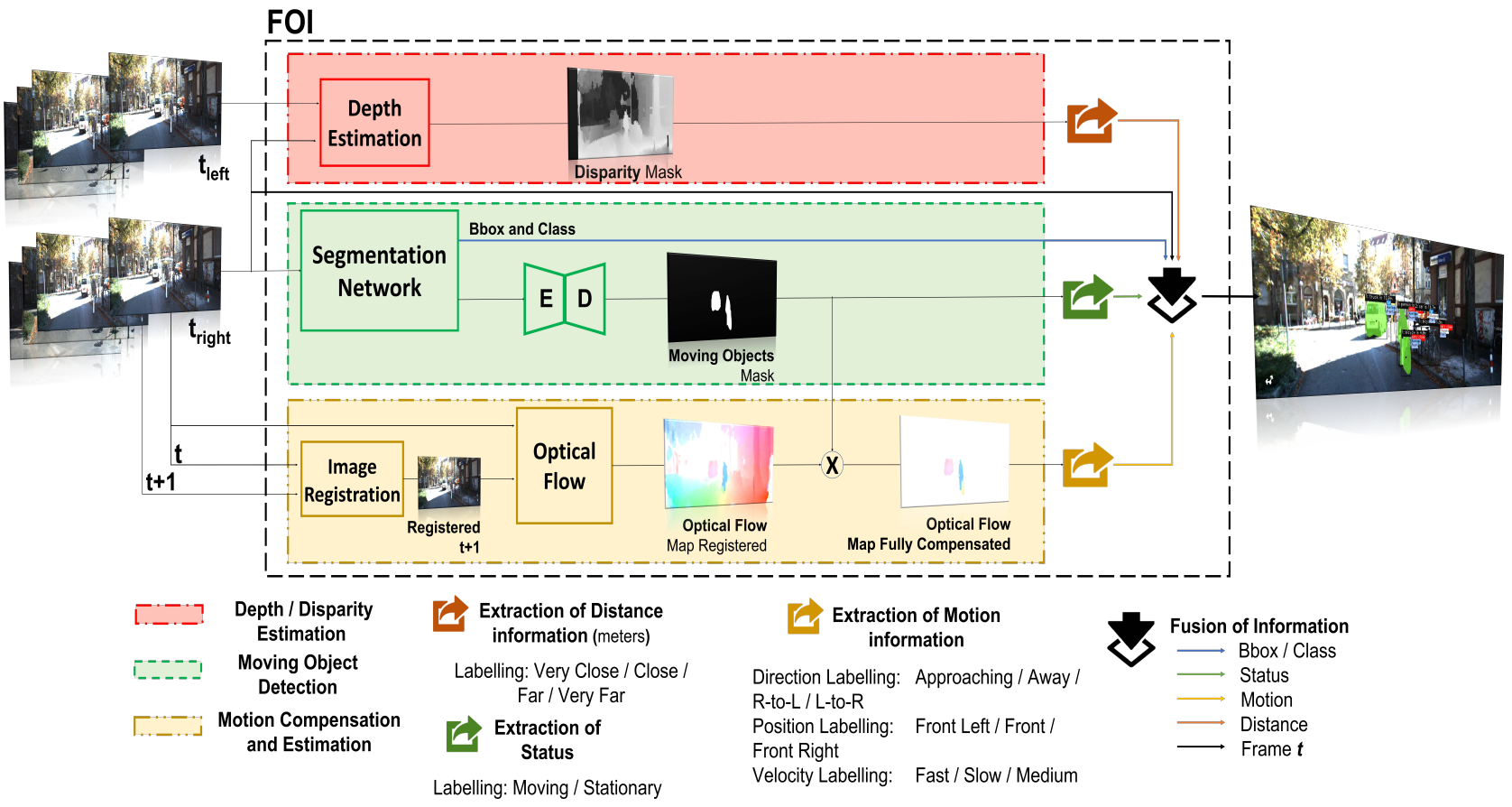


Figure 4.1: Structure and Work-flow of the proposed Framework for Object Identification (**FOI**)

4.2/ RELATED WORKS

In this section, we present the contributions of works that are most related to ours, i.e., scene understanding for driving by combining motion and geometry related information. These works mainly adopt moving object detection, motion segmentation, motion compensation for ego- motion, optical flow estimation, and depth estimation of stereo vision-based systems. A few works focused on object recognition or identification in a driving scene by combining various tasks among those mentioned above.

Some recent works have focused on hybridizing learning-based and geometry-based approaches. Chen et al. (2017c) proposed an approach to detect moving objects and estimate their motion states using sequential stereo images. The proposed system is a combination of several tasks; semi-global matching algorithm to compute disparity maps, and image segmentation is performed using simple linear iterative clustering (SLIC). The relationship between superpixels is sorted into coplanar, hinge and occlusion by applying slanted plane method. The motion of each superpixel is estimated based on the extracted feature points using RANSAC algorithm. Finally, superpixels with large possibilities of forming a single target and similar in motion are merged to extract moving objects. In Rateke and von Wangenheim (2020), the authors presented an approach to detect and recognize driving obstacles by combining multiple tasks, including image segmentation, depth estimation, and motion pattern extraction using optical flow. The method achieves good results by using depth and motion patterns. However, it cannot obtain the actual motion information of obstacles due to ego-motion. Li et al. (2020c) developed an image processing based system to identify various objects and predict the intention of pedestrians in the driving scene. The proposed model integrates multiple tasks including object detection, pose estimation, intent detection, dangerous vehicle detection, and traffic light detection. The proposed system uses the YOLOv4 model for object detection, skeleton-based intent detection for pedestrian pose estimation, and explainable artificial intelligence (XAI) technology is added for risk assessment (dangerous vehicle detection and traffic light detection).

Most of the existing work focused on the detection and tracking of moving objects in the driving environment. The authors in Cho et al. (2014) presented a multi-sensor fusion system for moving object detection and tracking in autonomous driving in urban environments. The proposed system has two parts; 1) sensor part composed of six radars, six LIDARs and three cameras. 2) fusion part where the measurements from different sensors are fused and presented according to their detection modalities (class, bounding box, distance, position, velocity and shape information of the objects). The system achieves promising results but requires multiple sensors which are quite expensive. Menze and Geiger (2015) proposed a model that estimates 3D scene flow using geometry and motion information of a small number of objects in the scene. This information (disparity

and optical flow) is extracted directly from active sensors. The authors introduced a new scene flow dataset with ground truth annotations for all static and dynamic objects in the scene. However, their approach is computationally expensive, and the proposed dataset is not large enough. In Siam et al. (2018b), the authors presented a Moving Object Detection Network (MOD -Net) model for autonomous driving that merges appearance and motion cues. They proposed from the existing KITTI dataset a new moving object detection dataset with weakly annotated segmentation masks (KITTI-MoSeg). Furthermore, Rashed et al. (2019) proposed a CNN (Fuse-MODNet) architecture for moving object detection by fusing RGB and LiDAR information. They provided an extended version of the KITTI-MoSeg dataset, the Dark-KITTI dataset, to simulate low-light driving environments. A real-time end-to-end CNN architecture for moving object detection (RST-MODNet) is presented in Ramzy et al. (2019). The proposed architecture benefits from temporal motion information embedded in sequential images and motion color maps using optical flow images.

By Yoo and Lee (2019), a moving object detection algorithm is developed using an object motion reflection model of motion vectors. The proposed method first generates the disparity map using stereo images and estimates the road by applying the v-disparity method of the disparity map. The motion vectors of symmetric pixels between adjacent frames are detected using optical flow (in which/where the road has been removed). They designed a probability model for how much local motion is reflected in the motion vector to determine if the object is moving. The authors Wu et al. (2020b) proposed a separate-predict-composite model for predicting future frames. Within the model, an encoder-decoder-based architecture for dynamic object detection is presented to identify objects between two classes, moving or static. The model takes multiple inputs (image sequences, semantic map, instance map and optical flow) and generates a binary mask to indicate the region of each moving object. Jung et al. (2020) proposed a foreground/background extraction based method for detecting moving objects from a moving camera using an inertial measurement sensor (IMU). The method used the Harris detector to extract points of interest, and epipolar geometry to classify the foreground (through the extracted map from image registration) and background feature points from successive images. Lee et al. (2020) developed a moving object detection and tracking method based on the interaction between Static Obstacle Map, which represents static obstacles, and Geometric Model-Free Approach for tracking moving objects, using point cloud information.

A few methods deal with the simultaneous estimation of the ego-motion of the vehicle and the motion of multiple moving objects in the scene. The authors in Vertens et al. (2017) propose an architecture for a semantic motion segmentation network (SMS-Net) that learns to predict both the semantic category and the motion state of each pixel from a pair of consecutive monocular images. They created their motion dataset (Cityscape-

KITTI-Motion), which contains over 3,155 manually annotated semantic motion labels. However, their method is very computationally intensive and cannot be applied to real-time. Moreover, their dataset primarily focused on vehicles only. Yu et al. (2019b) proposed an effective method for detecting moving objects using background subtraction. Global motion is estimated by tracking the grid-based key-points using optical flow. The authors Zhou et al. (2017) presented an approach for detecting moving objects from two consecutive stereo images. Their approach estimates the ego- motion uncertainty using the first-order error propagation model, which preserves the motion probability of each pixel. Pixels with similar depth and high motion probability are detected as moving pixels based on a graph-cut motion segmentation approach. However, the method is not robust to noise and unsuitable for real-world applications.

Our work is different in several aspects:

1. Most of the existing methods focused on detecting and tracking vehicles while ignoring the behavioral features related to movement, position, distance, and velocity.

2. Our proposed FOI is based on vision techniques that require data only from camera sensors, unlike existing methods, which used the combination of different sensors, i.e., camera, LIDAR, radar, inertial measurement unit, and other active sensors. The usage of multiple sensors is expensive, and it adds complexity and more challenges like multiple-sensor calibration, signal synchronization, and information association.

3. We proposed using image registration as a tool for ego-motion compensation due to the moving camera, then compute the optical flow to extract the moving objects' actual motion information in the driving scene.

4. Our approach is more generic than SOTA as we do not assume the object to be any specific type. The proposed FOI targeted objects covering vehicles, pedestrians, cyclists, motorcyclists, and others. The available moving object detection (MOD) datasets for the driving environment only consider moving vehicles while ignoring other critical dynamic objects mentioned above. Consequently, we developed an entirely new MOD dataset containing all of these dynamic elements.

The proposed framework extract all these features in order to allow better understanding of the driving scene.

4.3/ PROPOSED FRAMEWORK

In this section, we present the components of our object identification framework that extract accurate information about each object within the driving scene. These include depth

estimation using the semi-global matching algorithm, motion estimation using image registration and the optical flow method, and the moving object detection model (MOD). We also present the constructed motion-relevant annotations used to train the proposed MOD model. Finally, we will discuss how all the information are extracted and fused to understand the scene.

4.3.1/ DISPARITY/DEPTH ESTIMATION

Disparity is the distance between two pixel values or corresponding points in stereo pair images. The distance is calculated or estimated by comparing each pixel in the left image with the corresponding pixel in the right image.

$$Disparity(D) = X_{left} - X_{right} \tag{4.1}$$

where X_{left} and X_{right} are the same specific pixel coordinates in left, and right images, respectively, and D is the disparity value between these points. The depth (z-axis location point) can be calculated by using the disparity of the corresponding point Jain et al. (1995).

$$Depth(Z) = b * f / Disparity(D) \tag{4.2}$$

f is the focal length, and b is the baseline distance between the two cameras. The disparity map is a simple image representing pixel disparity values as an intensity image, the greater the intensity values, the higher the disparities or vice versa. The depth map image can be obtained by getting the depth of every pixel.

In this work, we adopt a well known Semi-Global Matching (SGM) algorithm Hirschmuller (2008), which calculates the matching cost (pixelwise), and aggregate these matching cost (from 2, 4, 8, or 16 paths) using equations 4.3 and 4.4:

$$L_r(p,d) = C(p,d) + min \begin{cases} L_r(p-r,d) \\ L_r(p-r,d+1) + p_1 - \min\limits_k L_r(p-r,k) \\ \min\limits_i L_r(p-r,i) + p_2 \end{cases} \tag{4.3}$$

where p is location of interest pixel, d is disparity value, $L_r(p,d)$ is cost path toward the actual pixel of path, $C(p,d)$ is pixel-wise matching cost, r is actual path and k is pixels in each path, p_1 and p_2 are the small and large values penalizing disparity changes between neighboring pixels of one pixel respectively.

$$S(p,d) = \sum_{r=1}^{2or4or8or16} L_r(p,d) \tag{4.4}$$

Then, by minimizing the aggregated cost values (equation 4.5), disparity for each pixel is calculated.

$$S = min_d S(p,d) \tag{4.5}$$

The SGM algorithm is faster than global matching algorithms and efficient compared to other methods Hirschmuller and Scharstein (2008). **Figure 4.2** show the disparity map example using SGM method. More details about SGM are given in the literature Hirschmuller (2008).

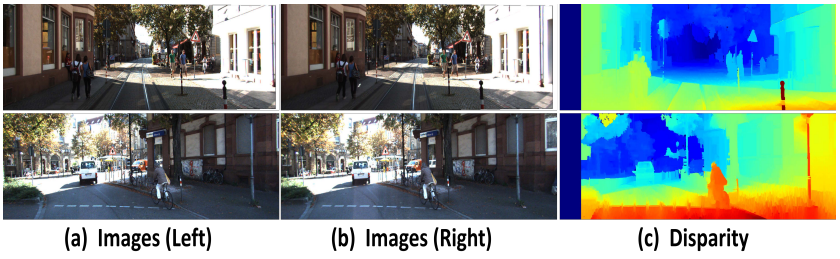


Figure 4.2: Disparity Map example on KITTI

4.3.2/ Motion Estimation

A vehicle may be driven in a driving scenario on different roads, at different speeds, daylight, conditions, seasons, and environments (e.g., urban, highway, and rural). Therefore, the situation is unpredictable while driving and is made more complex by the presence of moving objects in the scene, e.g., moving vehicles and pedestrians. The motion information of these moving objects is of great importance for safe driving in such scenarios. Numerous methods and techniques for extracting motion information have been studied and proposed. One of the most commonly used methods is optical flow estimation. It is expected that the accuracy of optical flow in the above scenarios or situations is good enough to ensure the reliability of the driving system. Optical flow based methods give satisfactory results when the camera is fixed or carefully displaced. However, the optical flow of image sequences captured by a moving camera encodes two pieces of information. The motion of the surrounding objects and secondly the motion of the ego vehicle result in significant motion vectors associated with the static objects, leading to a misperception of the static objects as moving objects. In this case, compensation of the camera motion is required. Therefore, we first compensate the ego-motion and later proceed with traditional optical flow method.

4.3.2.1/ Approach to Motion Compensation

Inspired by recent trends in aerial Zhang and Zhu (2020)and medical imaging Li et al. (2020a), we suggest a method called image registration for motion compensation. Image registration involves superimposing two or more images taken at different times, from different vantage points, and at different angles to obtain a 2D or 3D perspective. Various techniques are used for image registration such as wavelets, Fourier transform, correlation methods and feature based approaches. Image registration is done in four steps namely feature detection, feature matching, transformation model estimation, resampling of image and transformation. We used the image registration method presented in Forsyth and Ponce (2002) to compensate for the ego- motion of two consecutive images. The method relates different views of a scene via homographic transformations, finds and extracts features on one image (reference image), and matches them with the corresponding image (sensed image). Each considering pixel point (x, y) in the reference image and its corresponding pixel point $(\widehat{x}, \widehat{y})$ in the sensed image can be related through projective transformation

$$k\widehat{p} = H^T p \tag{4.6}$$

where $k \neq 0$ is an arbitrary scaling constant, $\widehat{p} = [\widehat{x}, \widehat{y}, 1]^T$, $p = [x, y, 1]^T$, and $H \in \mathbb{R}^{3\times3}$ with $H_{33} = 1$ is the unknown projective transformation matrix. Given degree of freedom $d > 3$ correspondences $\left\{(x_i, y_i) \rightarrow (\widehat{x_i}, \widehat{y_i})\right\}_{i=1}^{d}$,

H can be estimated in a least squares sense Forsyth and Ponce (2002),

$$min_h ||Ah||^2 \; s.t. h_9 = 1 \tag{4.7}$$

where $h = vec(H)$ is the vectorized version of H formed by stacking its columns into a vector, $A^T = [A_1^T, ..., A_d^T]$, with

$$A_i = \begin{bmatrix} 0 & p_i^T & -\widehat{y_i} p_i^T \\ p_i^T & 0 & -\widehat{x_i} p_i^T \end{bmatrix} \in \mathbb{R}^{2\times9} \tag{4.8}$$

The solution of the equation 4.7 is the smallest right singular vector of A, scaled so that the last element is 1.The method adopts the Binary Robust Invariant Scalable Keypoints (BRISK) algorithm Leutenegger et al. (2011) to compute features (multi-scale corner features) from the reference and sensed images. Then, the Random Sample Consensus (RANSAC) algorithm Fischler and Bolles (1981) is used to find a robust subset of the correspondences that yield a solution $\widehat{H}$ of the equation 4.7. RANSAC is designed to remove the outliers from the matching features and keep only the correct matches, which

are used to estimate the registration parameters. The nature of the transformation is projective. We finally obtain a registered image in which the background becomes stable, and we call it compensated background image. **Figure 4.3** shows the workflow of image registration.

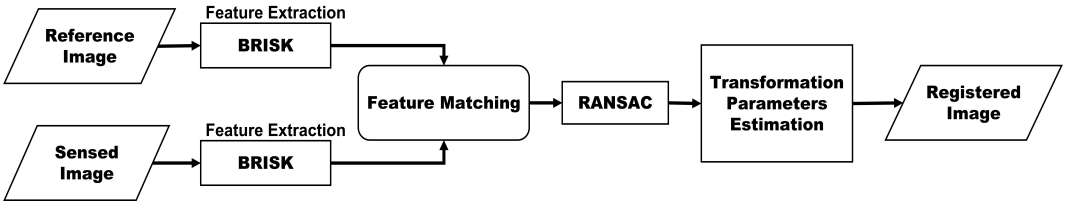


Figure 4.3: Work Flow of Image Registration

4.3.2.2/ COMPUTING OPTICAL FLOW

Optical flow calculates the approximation to the motion field from the change in intensity of the image during a given time frame. It can be visualized in arrows (motion vector, which provides an excellent intuitive perception of the physical motion) or color (hue providing the direction and saturation giving vector magnitude).

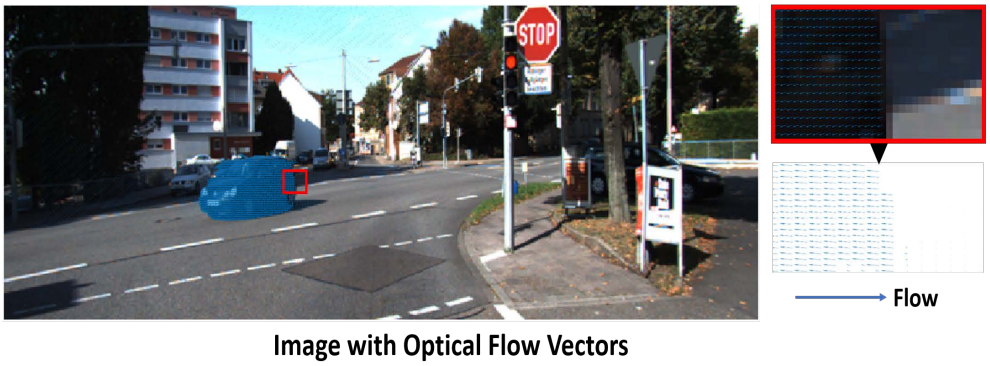


Figure 4.4: Example of optical flow with flow-vectors

We use a traditional state-of-the-art optical flow algorithm Farnebäck (2003), which extracts motion vectors using information obtained from two consecutive frames. In **Figure 4.4**, an example image with flow vectors is shown, a car moves from right to left in front of a reference vehicle waiting at a traffic light. The algorithm derives flow vectors containing both the directions and magnitude of the pixel motion, which are later used to extract information such as direction, position, and velocity.

We propose the use of image registration along with optical flow to overcome ego- motion and obtain an actual estimate of the motion information. First, the image registration algorithm is applied to two consecutive images (t and $t+1$) to obtain a registered image, as shown in the orange block of Figure 4.1. Then, the optical flow is calculated using the

image (t) and the obtained registered image ($t+1$). The output of this procedure is called the registered optical flow map. The effects of image registration on the estimated optical flow, flow vectors, and flow color maps are shown in **Figures 4.5** and **4.6**, respectively. Image registration has a significant impact on optical flow but cannot fully compensate for camera motion. Some static objects are still represented as moving objects such as poles, traffic lights, buildings, trees, etc. To overcome this problem, we propose a deep neural network model for moving object detection based on a segmentation network and an encoder-decoder network (detailed in 4.3.3). The results of the proposed network and the registered optical flow are fused to fully compensate the ego-motion (4.3.4).

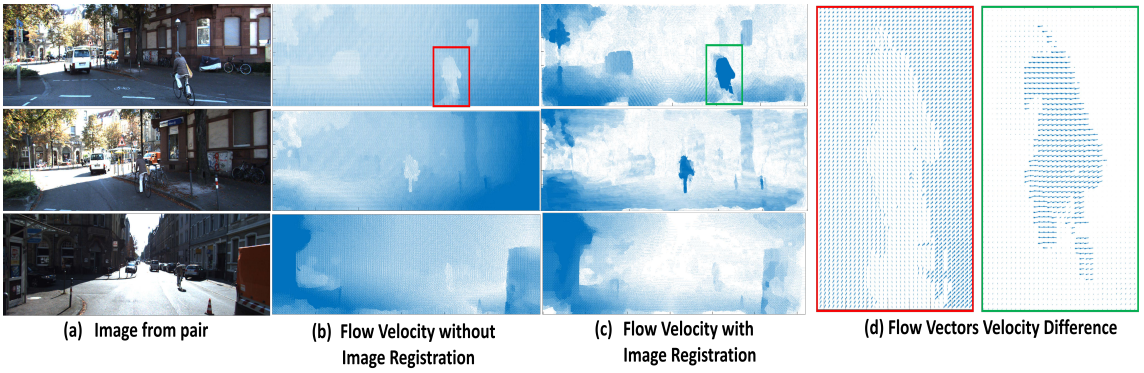


Figure 4.5: Left to right: (*a*) First image from a pair. (*b*) Flow vectors without image registration. (*c*) Flow vectors with image registration. (*d*) Flow Velocity difference

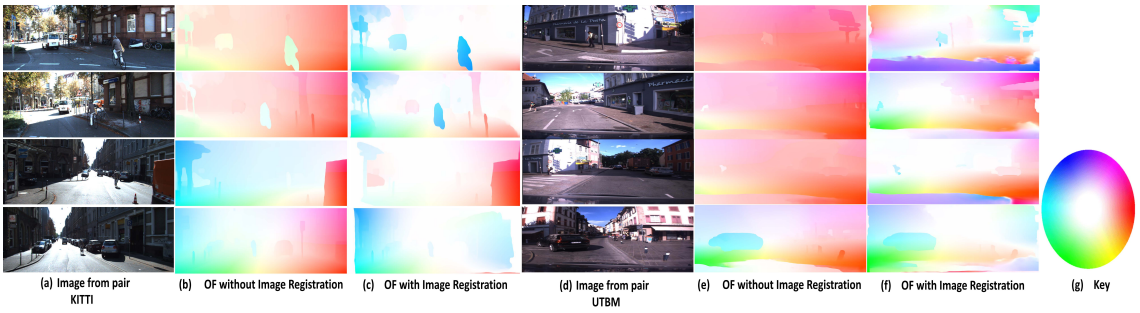


Figure 4.6: Left to right: (*a*) First image from a pair KITTI. (*b*) Corresponding computed optical flow without image registration. *c*) Optical flow with image registration. (*d*) First image from a pair UTBM. (*e*) Optical flow without image registration. *f*) Optical flow with image registration.(*g*) Key: color map to display flow field

4.3.3/ PROPOSED MOVING OBJECT DETECTION MODEL

The more accurate and practical way to detect a moving object in vision tasks is to understand the motion over two or more successive images. We used a simple moving object detection idea to first detect the interesting objects and then classify the moving ones. Our approach involves two mutual tasks: Object segmentation of certain classes such as

Table 4.1: **SOTA Instance segmentation networks detection performance on COCO dataset** test-dev2017

Method	Backbone	Mask AP	Box AP	fps	GPU
Mask R-CNN Wu et al. (2019a)	R-50-FPN	35.2	38.6	23.2	V100
Mask R-CNN Wu et al. (2019a)	R-101-FPN	38.6	42.9	17.8	V100
CenterMask Lee and Park (2020)	R-101-FPN	39.8	44.0	15.2	V100
CenterMask-Lite Lee and Park (2020)	V-39-FPN	36.3	40.7	35.7	Xp

vehicles, pedestrians, bicyclists, and motorcycles. The second task is binary pixel classification, which uses temporal information to predict whether the detected object is moving or static.

SEGMENTATION NETWORK

We used an instance segmentation network in the object segmentation task, which provides the segmentation mask, bounding boxes, and category probabilities for each object of interest. We incorporate segmentation network **Mask R-CNN** He et al. (2017a) with different backbone architectures into our framework for the object segmentation task. We used the model implemented by Wu et al. (2019a) based on Feature Pyramid Network (FPN), ResNet-50 and ResNet-101 backbone trained on MSCOCO Lin et al. (2014b) dataset. We also explore another more effective and faster instance segmentation network with two different backbone architectures (ResNet-101-FPN, VoVNetV2) recently proposed by Lee and Park (2020), called **CenterMask**. We used these networks for the following reasons: They are state-of-the-art instance segmentation networks with the highest classification accuracy and high speed. **Table 4.1** shows their detection performance on the COCO dataset reported in Wu et al. (2019a)Lee and Park (2020).

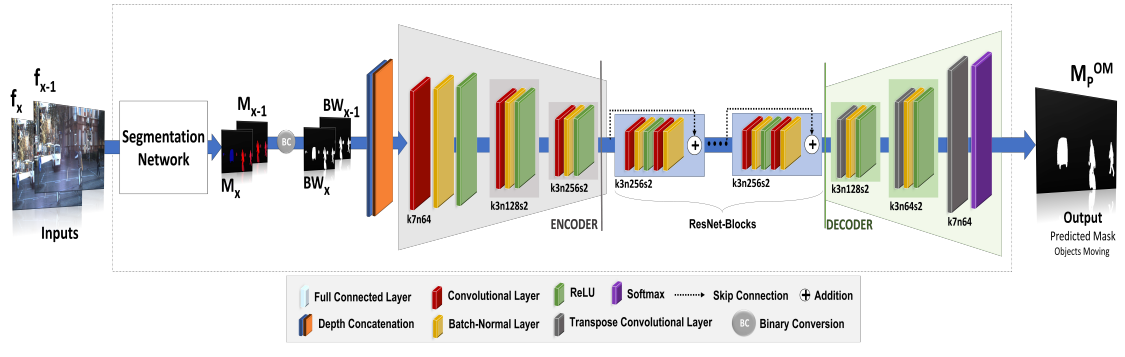


Figure 4.7: Structure and Flow of two mutual tasks for moving object detection (MOD).

ENCODER-DECODER NETWORK

The task of temporal processing is performed by an encoder-decoder network (**EDNet**) that classifies only moving ones using segmented masks of consecutive frames from the object segmentation task. The EDNet is based on the well-known deep ResNet He et al. (2016). The ResNets have been tested in many benchmarks and have achieved significant performance. The ResNet structure has a set of residual blocks in which information is propagated by skip-connection (bypassing the nonlinear layers). We have embedded three down-sampling blocks for encoding in the ED network, residual blocks that extract discriminative features, and three up-sampling blocks as decoder parts. All the encoding blocks and the first two decoding blocks include two-stride convolution/deconvolution, batch- normalization (BN), and rectified Linear Unit (ReLU). Residual blocks consist of the structure of convolutional layers, batch normalization, ReLU, convolutional layers, and batch normalization. The last up-sampling block consists of transposed convolutional layer and softmax activation layer. **Figure 4.7** illustrates the structure and flow of two mutual tasks for moving object detection. The input of EDNet is the concatenation of two consecutive masks (temporal information) of objects of interest generated by segmentation network. The masks contain both moving and static objects of the scene. The EDNet then further classifies them and extracts only the moving objects using back-propagation training according to the ground truths.

4.3.3.1/ MOD DATASETS

There are few datasets for detecting moving objects in a driving environment. The existing publicly available MOD datasets focused only on vehicles with object categories of cars, trucks and vans (summarized and compared in **Table 4.2**). KITTI-Motion contains 273 training and 230 test images, while 1300 training and 349 test images are provided for KITTI-MoSeg. The extended KittiMoSeg dataset offers more than $12k$ binary mask labels (10222 training and 2697 test images) for different sequence runs from the KITTI dataset. However, there are about 7k labels that do not contain moving objects. Many labels are ambiguous, i.e., objects are labeled in a square area, incorrect labeling of moving objects, etc. For this reason, we manually selected only the image labels where the moving objects are correctly labeled (4800 for training and 1927 for testing).

Our goal is to detect all types of moving objects in the scene. Therefore, we developed a large moving object detection dataset that covers all dynamic objects such as all types of vehicles, pedestrians, cyclists, motorcyclists, buses, trains, and trucks.

PROPOSED MOD DATASET

Our idea for moving object detection is to first detect the objects of interest that may exhibit motion, and then identify the moving ones among them. We have adopted the mask R-CNN segmentation model which generates the segmentation masks for each instance of an object in the image. We build our new dataset using these generated segmentation masks. The object segmentation step is considered as data pre-processing/preparation for the temporal processing step. An overview of the dataset preparation flow can be found in the **Figure 4.8**. From the segmentation masks, we quickly obtain the masks of the objects of interest (c). Next, we manually annotated the masks of objects of interest for relevance to object motion from sequence frames (manually identifying objects from multiple frames and keeping moving). We used different sequences from the KITTI raw dataset Geiger et al. (2015) and EU long term dataset Yan et al. (2020) to create a total of 10059 semantic segmentation mask images (with static/moving objects of interest) with corresponding annotated binary mask labels (with moving objects only). Each binary mask label for moving objects is created from the corresponding sequence pair images. Table 4.2 shows a summary comparison of our dataset with existing available MOD datasets.

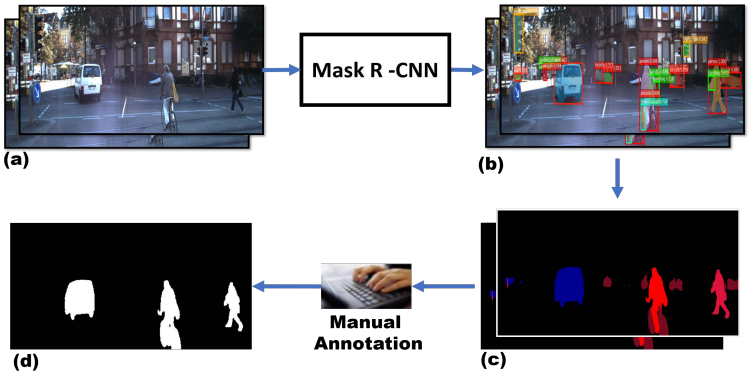


Figure 4.8: Flow for generating motion relevant annotations. (a) input image (b) model generate bounding boxes and segmentation masks for each instance of an object in the frame (c) objects of interest mask (d) manually annotated moving objects mask

Table 4.2: **Comparison with existing available Moving Object datasets**

Datasets	No of Images	Object Classes	Course	Image Resolution
KITTI-Motion Vertens et al. (2017)	455	Vehicles Only	✓	384×1048
Cityscapes Motion Vertens et al. (2017)	3475	Vehicles Only	✗	384×768
KITTI-MoSeg Siam et al. (2018b)	1300	Vehicles Only	✓	384×1280
KITTI-MoSeg Extended Rashed et al. (2019)	12919	Vehicles Only	✓	1024×2048
Ours	10059	All type Vehicles Pedestrians Cyclists Motor bike	✓	375×1242

Algorithm 1: Training process

Input: f_x and f_{x-1} : Two consecutive frames
Output: M_P^{OM} : Moving objects mask predicted
Functions: $\mathbb{SEG}()$: load trained segmentation model with weights freezing ;
for N epochs **do**
  for N/m steps **do**
    sample a mini-batch of m two consecutive frames [$\{f_x, f_{x-1}\}$, ... ,$\{f_y, f_{y-1}\}$] $(x, y) \in [1, N_f]$;
    calculate the corresponding segmentation masks [$\{M_x, M_{x-1}\}$, ... ,$\{M_y, M_{y-1}\}$] using the $\mathbb{SEG}()$:
    $\{M_x, M_{x-1}\}$=$\mathbb{SEG}(\{f_x, f_{x-1}\})$;
    convert M_x and M_{x-1} to binary BW_x and BW_{x-1} ;
    concatenate BW_x and BW_{x-1} on the channel dimension $BW_{x \bigcup x-1}$;
    feed $BW_{x \bigcup x-1}$ to the ED-Net to calculate the moving objects mask M_P^{OM} ;
    calculate the descending Cross Entropy loss of the ED-Net:
    $-\sum_i^2 M_{GT}^{OM} \log M_P^{OM}$;
    update the ED-Net parameters using Adam optimizer ;

4.3.3.2/ MOD TRAINING

The training procedure is summarized step-by-step in **Algorithm 1**. The segmentation network (Figure 4.7) generates the segmentation masks of the objects of interest for each frame (t), which is combined with the previous frame (t_{i-1}), and both are fed to the EDNet, which helps the EDNet to learn the temporal relationships between the pixels and use the relationships to predict the motion class. The first step in the EDNet is a depth concatenation layer, which takes as input the binarized image masks (t_i) and (t_{i-1}) and concatenates them along the third dimension before entering the first downsampling block, which consists of a convolution with a kernel size of 7 and a feature map size of 64, followed by an element-wise batch normalization layer and a ReLU operation. Next, the two remaining downsampling blocks are executed with a kernel size of 3 and a stride of 2. Then, the ED-Net extracts more learnable features through ResNet blocks (3, 6 and 9), each containing 5 operations. In the decoder, the first two blocks use a transposed convolution with a kernel size of 3, batch normalization, and ReLU layers for upsampling feature maps before running through the final transposed convolution with a kernel of 7 and softmax layers. We chose the cross-entropy loss function to fit the predicted probability distribution (q) to the ground truth (true distribution p).

$$H(p, q) = -\sum_{i=1}^{Nc} p(i) \log q(i) \tag{4.9}$$

where Nc is the number of classes, which is in our case equal to two since we want to classify moving and static objects.

4.3.3.3/ MOD EXPERIMENTS

We first start with the metrics used for the evaluation. Then, we present the proposed Moving Object Detection model, training and testing parameter, proposed MOD dataset, evaluation and comparison with state-of-the-art methods for moving object detection on existing MOD datasets.

EVALUATION METRICS

We evaluate our MOD model using different metrics; a standard mean intersection over union ($mIoU$) metric, Precision/Recall, and F1 score.

IoU: Intersection over union can be computed for class as follows

$$IoU = \frac{TP}{(TP + FP + FN)} \tag{4.10}$$

where TP, FP and FN correspond to true positives, false positive and false negative respectively. Then, the $mIoU$ is the average of the computed $IoUs$ regarding the number of classes

$$mIoU = \frac{1}{N_c} \sum_{i=1}^{N_c} IoU_i \tag{4.11}$$

as N_c is the number of classes and IoU_i is the Intersection over union calculated for i^{th} class.

Precision: Describes the purity of positive detections relative to the ground truth

$$Precision = \frac{TP}{(TP + FP)} \tag{4.12}$$

Recall: Describes the completeness of our positive predictions relative to the ground truth

$$Recall = \frac{TP}{(TP + FN)} \tag{4.13}$$

F1 Score: It is the harmonic mean of the precision and recall

$$F1_{Score} = \frac{TP}{(TP + \frac{1}{2}(FP + FN))} \tag{4.14}$$

Table 4.3: Quantitative evaluation of our MOD model on the validation set of the proposed MOD dataset. Comparison of different design variants for segmentation (Mask-RCNN and CenterMask) and Encoder-decoder network (with 3, 6 and 9 residual blocks). The evaluation is in the form of intersection over union, precision, Recall, F-score and frame per second, using the respective image resolutions.

Approach	Segmentation Method used	Backbone	N Image Validation	mIoU	Moving IoU	Static IoU	Precision	Recall	Fscore	fps
ResNet 3-Block	Mask R-CNN	R-50-FPN	1509	82.48	66.07	98.91	72.54	77.43	73.07	9.309
	Mask R-CNN	R-101-FPN		83.16	67.37	98.95	73.70	77.32	75.52	8.199
	CenterMask	R-101-FPN		83.60	68.24	98.97	74.79	77.42	76.79	8.152
	CenterMask-Lite	V-39-FPN		82.98	66.09	98.91	73.28	77.39	73.99	9.990
ResNet 6-Block	Mask R-CNN	R-50-FPN	1509	84.36	69.99	98.74	76.90	78.01	76.39	9.265
	Mask R-CNN	R-101-FPN		84.46	71.99	98.65	77.41	78.6	76.48	8.174
	CenterMask	R-101-FPN		85.58	73.18	98.99	77.95	78.71	76.68	8.092
	CenterMask-Lite	V-39-FPN		84.41	71.06	98.75	76.03	78.03	76.4	10.006
ResNet 9-Block	Mask R-CNN	R-50-FPN	1509	83.15	67.47	97.94	74.42	77.20	75.37	9.201
	Mask R-CNN	R-101-FPN		84.04	70.9	98.77	75.13	78.01	75.91	8.098
	CenterMask	R-101-FPN		84.84	71.83	97.51	75.49	78.27	75.15	8.003
	CenterMask-Lite	V-39-FPN		83.33	69.91	98.95	73.78	77.18	75.59	9.998

SEGMENTATION NETWORK AND EDNET ADOPTION

We performed an ablation study with the different numbers of ResNet residual blocks, i.e., 3, 6, and 9 blocks, together with four segmentation model choices (Mask-RCNN R50/101, CenterMask-R101/V39) to observe the trade-off between accuracy and speed. We trained our MOD model on the proposed moving object dataset and evaluated its performance. We split the annotated images into 85% (8550) and 15% (1509) for the training set and validation set, respectively. Out of the total 10059 mask images, 6249 masks have moving objects, and the remaining 3810 masks have no moving object (black image). We also use masks without moving objects during training, which helps the model to understand the appearance of static objects as well and also reduces the over-fitting. The evaluation is performed on images with resolution of 1242×375 on Nvidia GTX-2080Ti GPU. We obtain promising results (in terms of accuracy) for all segmentation models within our MOD, trained on the proposed dataset. **Table 4.3** show the metric scores of our MOD model on the newly proposed dataset using different segmentation model and ResNet blocks architectures within the proposed MOD. The Mask R-CNN (R101-FPN) and CenterMask (R101-FPN) segmentation models with six block ResNet achieve high accuracies with low processing time. Therefore, we decide to use CenterMask-Lite (V39-FPN) segmentation model and six ResNet blocks approach for our proposed FOI framework as it achieves good accuracy at high speed.

Figure 4.9 presents the qualitative results of our MOD model on the proposed dataset with complex scenes. It can be seen that the model accurately segments the moving objects including vehicles, pedestrians, cyclists and motorcyclists in different sequence passes.

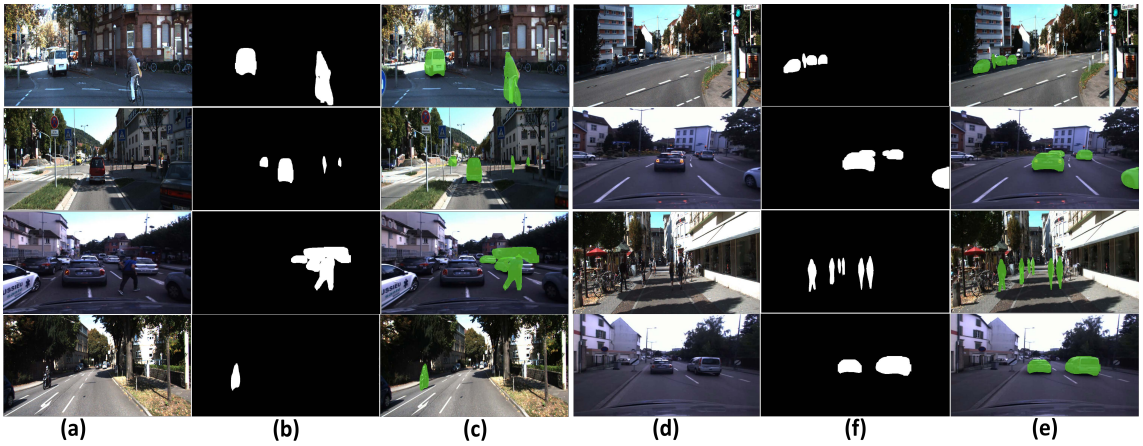


Figure 4.9: Our MOD model results on the proposed MOD dataset. (a) and (d) are input images from sequence pair, (b) and (f) are predicted moving object masks, and (c) and (e) are the overlap of the mask on the image.

MOD Comparison against SOTA

The proposed model labels the moving objects (as white) and the static/background (as black) from sequence pair images. This allows us to compare our results with other state-of-the-art methods that treat moving object detection as a pixel-wise binary segmentation problem. Training and evaluation of our proposed MOD model have been performed individually on the following datasets: the proposed MOD dataset, KITTI-Motion Vertens et al. (2017), KITTI-MoSeg Siam et al. (2018b), and KITTI-MoSeg Extended Rashed et al. (2019). The evaluations include a quantitative and qualitative comparison on these four datasets and a qualitative one on sequence images from the KITTI benchmark. We cannot train SOTA-MOD models on our proposed dataset for the following reasons: Unavailability of source code, the existing method is based on multiple input sources, e.g., lidar information, optical flow map, Inertial Measurement Unit (IMU) and odometry data, etc. The figures for the SOTA methods are from the respective original papers. The SOTA methods had results on public datasets, which are different from the proposed dataset (the existing public datasets focus only on vehicles and not on all types of moving objects). The proposed dataset covers all moving objects in the current public datasets, so we believe it is fair to compare the presented MOD quantitative results with the SOTA methods.

Comparison against SOTA methods on KITTI-Motion: The **Table 4.4** highlights the performance of our method on KITTI-Motion Vertens et al. (2017) in terms of mean intersection over union (mIoU), running frames per second (fps) on image resolution 384×768, and testing GPU. The scores of SOTA methods are taken from the reference papers Ramzy et al. (2019)Siam et al. (2018a). We outperform all the methods on the KITTI-Motion dataset based on overall IoU. We cannot reasonably compare the model's speed

results, as different GPUs are used in testing. Qualitative results are illustrated in **Figure 4.10**, comparing the proposed method against SmSNet Vertens et al. (2017) and RTMotSeg Siam et al. (2018a) on KITTI-Motion Vertens et al. (2017) dataset.

Table 4.4: Quantitative comparison against state-of-the-art methods on KITTI-Motion Vertens et al. (2017) dataset.

Approach	mIoU	fps	GPU
GEO-M Kundu et al. (2009)	48.15	-	-
AHCRF+Motion Lin and Wang (2014)	68.0	-	-
RTMotSeg (RGB+Mot) Siam et al. (2018a)	68.8	25	Titan X Pascal
MODNet Siam et al. (2018b)	72	6	Titan X Pascal
RTMotSeg (RGB+Mot+PrpModel) Siam et al. (2018a)	72.3	17	Titan X Pascal
RST-MODNet Ramzy et al. (2019)	83.7	21	Titan X Pascal
SmSNet Vertens et al. (2017)	84.1	7	Titan X Pascal
Ours	**85.33**	10	RTX-2080Ti

Figure 4.10: Qualitative comparison against SmS-Net Vertens et al. (2017) and RTMotSeg Siam et al. (2018a) on KITTI-Motion.

Comparison against SOTA methods on KITTI-MoSeg: In **Table 4.5**, we compared our results against MODNet Siam et al. (2018b) and U^2-ONet Wang et al. (2021a) on KITTI-MoSeg Siam et al. (2018b). It can be seen that the proposed model outperforms MODNet and U^2-ONet in all metrics (moving IoU, precision, recall, and F_{score}). In terms of moving IoU, our model outperforms MODNet and U^2-ONet by 13.41% and 3.35%, respectively. The proposed MOD runs at 10 fps on an RTX-2080Ti GPU for an input image size of 384×1048, which is higher than MODNet (8 fps). Qualitative comparisons covering the proposed model and MODNet are shown in **Figure 4.11**.

Comparison against SOTA methods on KITTI-MoSeg Extended: We highlight the performance of the proposed MOD model as compared to Fuse-MODNet Rashed et al. (2019), RST-MODNet Ramzy et al. (2019), and U^2-ONet Wang et al. (2021a) trained on KITTI-MoSeg Extended Rashed et al. (2019) dataset in **Table 4.6**. The authors in Rashed

et al. (2019) and Ramzy et al. (2019) proposed architectures that support different inputs from different sensors. We compared our results with all their input configurations. The evaluation is performed with input image resolution 384×1280. We have significantly outperformed all Fuse-MODNet and RST-MODNet sensors fusion methodologies and U^2-ONet in terms of mIoU and Moving IoU. **Figure 4.12** shows the qualitative assessment between the proposed model, Fuse-MODNet Rashed et al. (2019), RST-MODNet Ramzy et al. (2019), and U^2-ONet Wang et al. (2021a). The extended KittiMoSeg dataset provides more than $12k$ binary mask labels for different sequence runs from the Kitti dataset. However, there are approximately 7k labels that do not have moving objects. Many labels are ambiguous, i.e., objects are labeled in a square area, incorrect labeling of moving objects, etc. For this reason, we manually selected only those image labels where the moving objects are labeled accurately (4800 for training and 1927 for testing).

4.3.4/ Motion Compensation - Fully Compensated Optical Flow

We integrate the optical flow results with the moving object detector to obtain the flow map for moving objects only, which we call the Fully Compensated Optical Flow (**FCOF**) color map, as shown in **Figure 4.13**. So we can say that we fully compensate the moving camera's ego-motion from these resulting flow maps. The sought motion information such as direction, position and velocity are extracted from the specific pixel values of each object, which allows us to create a detailed motion analysis for each object in the driving scene.

Table 4.5: Quantitative comparison on KITTI-MoSeg Siam et al. (2018b) dataset

Approach	Moving IoU	Precision	Recall	Fscore	fps	GPU
MODNet Siam et al. (2018b)	45.41	56.18	70.32	62.46	8	Titan Xp
U^2-ONet Wang et al. (2021a)	55.47	68.08	72.36	64.23	-	Tesla V100
Ours	**58.82**	**70.83**	**76.87**	**70.23**	10	RTX-2080Ti

Figure 4.11: Qualitative comparison against MODNet Siam et al. (2018b) on KITTI-MoSeg.

Table 4.6: Quantitative comparison on KITTI-MoSeg Extended Rashed et al. (2019) dataset

Approach	mIoU	Moving IoU	fps	GPU
Fuse-MODNet Rashed et al. (2019) (RGB)	65.6	32.7	40	Titan Xp
Fuse-MODNet Rashed et al. (2019) (RGB+rgbFlow)	74.24	49.36	25	Titan Xp
Fuse-MODNet Rashed et al. (2019) (RGB+lidarFlow)	70.27	41.64	25	Titan Xp
Fuse-MODNet Rashed et al. (2019) (RGB+rgbFlow+lidarFlow)	75.3	51.46	18	Titan Xp
RST-MODNet Ramzy et al. (2019) (LSTM-Multistage)	76.3	53.3	23	Titan Xp
U^2-ONet Wang et al. (2021a)	-	62.5	-	Tesla V100
Ours	**80.15**	**64.11**	10	RTX-2080Ti

4.3.5/ Fusion of MOD, FCOF and Disparity

The results of each stage of the proposed framework, such as disparity, moving object detection, and motion estimation, are fused to extract information such as object ID, static or moving, distance, direction, position, and velocity. The pseudo-code for the information extraction is given in **Algorithm 2**.

Figure 4.12: Qualitative comparison against Fuse-MODNet Rashed et al. (2019), RST-MODNet Ramzy et al. (2019), and U^2-ONet Wang et al. (2021a) on KITTI-MoSeg Extended.

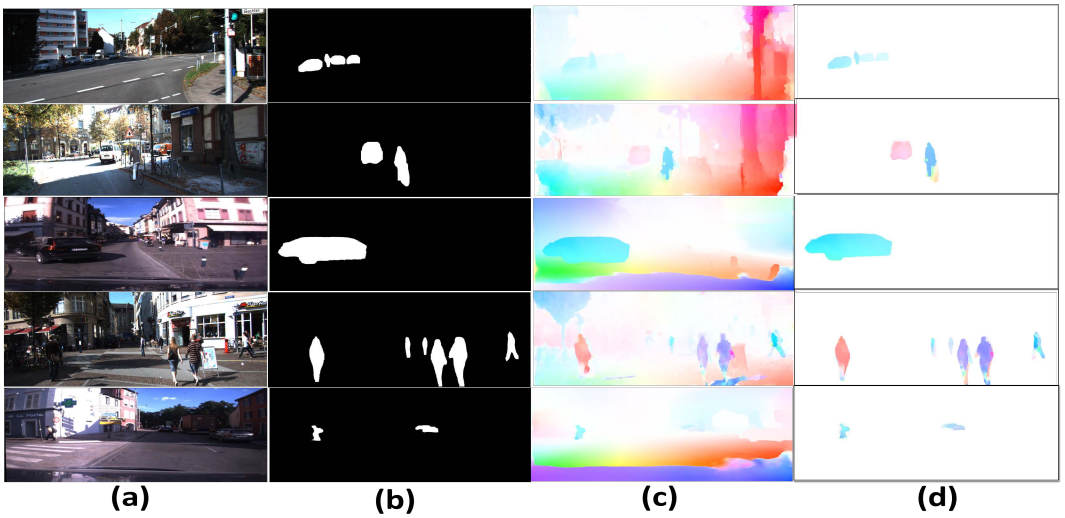


Figure 4.13: Left to right: (*a*) Frames (*b*) Detected moving objects masks by (MOD) Model (*c*) registered Optical flow maps after image registration (*d*) Fully compensated optical flow color maps (combining (b) and (c))

DIRECTION

The optical flow map gives polar coordinates of motion direction and intensity for each pixel of the detected object. We compute the average motion values by finding their mean values for the exact direction and motion intensity. The direction values can be calculated from the motion vector or color map (angle to direction and magnitude to velocity). For example, from the motion vector on the x axis for labeling if the object is motionless/static $(-1 = 1)$. Similarly on the y axis for labeling if the object is moving away $(y <= 1)$ or approaching $(y <= -1)$, or is motionless $(-1 < y < 1)$.

POSITION

The direction was discretized from the viewpoint of the target vehicle (see Figure 4.14). The relative position of each object can be defined by the object ”front left”, ”front” or ”front right”.

VELOCITY

Velocity can be calculated from vector values representing the displacement of a pixel between two frames. The displacement values or intensity values from each object are collected by multiplying the mean values of the x axis and y axis from each object. These intensity values represent the speed of movement and are labeled as very fast, fast, medium, slow, very slow, and stationary.

Algorithm 2: Pseudocode for extraction of Motion Direction, Position, Distance, and Velocity information.

Inputs: M_{dprty} ▷ Disparity Mask M_P^{OM} ▷ Moving Object Mask BC_P^{OI} ▷ Bounding box, class information of objects of interest from MRC V_{uv}^{OF} ▷ Fully Compensated Optical flow vectors V_x and V_y

Outputs: f_P^{OI} ▷ Frame$_{pred}$ - Objects Identified $json_{file}$ ▷ json file - Information of each object in each frame

Functions:

$\mathbb{DC}()$: returns average intensity from M_{dprty}; $\mathbb{DRC}()$: returns direction from V_{uv}^{OF};

$\mathbb{VC}()$: returns velocity ; $\mathbb{PC}()$: returns position ;

$\mathbb{DC}() \leftarrow M_{dprty}$

Step 1: Find Contours on the M_{dprty}

Step 2: Calculate the distances to the contour

Step 3: Calculate average intensiites (DI_{avg})

Step 4: Calculate distance in meters S_{dis} using equation 4.2

if $S_{dis} \geq 0.18$ **then** $label_{dis}$ = Very close

elif $0.12 \leq S_{dis} \leq 0.179$ **then** $label_{dis}$ = Close

elif $0.05 \leq S_{dis} \leq 0.119$ **then** $label_{dis}$ = Far

elif $S_{dis} \leq 0.049$ **then** $label_{dis}$ = Very far

$\mathbb{DRC}() \leftarrow V_{uv}^{OF}$

Calculate averages (Vx_{avg} & Vy_{avg})

if $Vx_{avg} \geq 1$ **then** $label_{Hdir}$ = Left to Right

elif $Vx_{avg} \leq -1$ **then** $label_{Hdir}$ = Right to Left

elif $-1 < Vx_{avg} < 1$ **then** $label_{Hdir}$ = Motionless

if $Vy_{avg} \geq 1$ **then** $label_{Vdir}$ = Approaching

elif $Vy_{avg} \leq -1$ **then** $label_{Vdir}$ = Moving Away

elif $-1 < Vy_{avg} < 1$ **then** $label_{Vdir}$ = Motionless

$\mathbb{VC}() \leftarrow Vx_{avg}$ & Vy_{avg}

Calculate vector length ($V_{Len_{HV}}$)

$V_{Length_{HV}} = Vx_{avg} \times Vy_{avg}$

if $V_{Len_{HV}} \geq 90$ **then** $label_{velocity}$ = Very Fast

elif $9 \leq V_{Len_{HV}} < 90$ **then** $label_{velocity}$ = Fast

elif $0.9 \leq V_{Len_{HV}} < 9$ **then** $label_{velocity}$ = Medium

elif $0.09 \leq V_{Len_{HV}} < 0.9$ **then** $label_{velocity}$ = Slow

elif $V_{Len_{HV}} < 0.09$ **then** $label_{velocity}$ = Stationary

$\mathbb{PC}() \leftarrow frame_{width}$

if $w < frame_{width}/3$ **then** $label_{pos}$ = Front Left

elif $w < frame_{width} \times 0.66$ **then** $label_{pos}$ = Front

else $label_{pos}$ = Front Right

In the predicted f_P^{OI} frame output, show the object ID/class on top of binding box $\leftarrow BC_P^{OI}$, direction arrow $\leftarrow$ ($label_{Hdir}$ & $label_{Vdir}$) in the center, while distance $\leftarrow$ ($label_{dis}$), position $\leftarrow$ ($label_{pos}$) and velocity $\leftarrow$ ($label_{velocity}$) on top-right of the binding box of each object.

Label moving objects $\leftarrow M_P^{OM}$ in green on the frame f_P^{OI}.

Generate json file $json_{file}$, containing each detected object information for identification.

DISTANCE

We compute the disparity and depth values (intensities) using the (SGM) algorithm and calibrate it to roughly visualize the distance of each segmented object according to its average intensity value. We define four labels for depth: very far, far, close, and very

close.

LABELLING ANS SCALING

The extracted information of each object is labeled and scaled in different colors, using the mapping in Figure 4.14 for visual representation. Also, the extracted information details of each detected object in the image are stored in a json file (later used for evaluation).

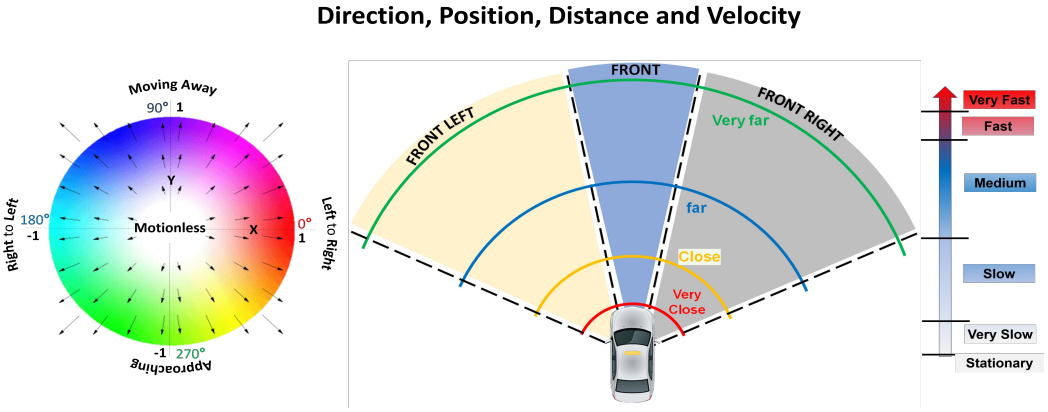


Figure 4.14: Qualitative Mapping

4.4/ EXPERIMENTAL ANALYSIS

This section is divided into two part. In the first part, the evaluation of each task is presented, such as object status (moving/static), object motion, position, and distance. In the second part, we present the experimental results of the whole framework of object identification (FOI). We quantitatively demonstrate the accuracy of FOI and compare the results of FOI with manually annotated Object-wise Semantic Information (**OSI**).

OBJECT-WISE SEMANTIC INFORMATION (**OSI**) ANNOTATIONS

There is no single standard format for image annotations, different datasets provide different annotation formats, e.g. COCO stores annotations in JSON, Pascal VOC in XML files, etc. In this work, we create **.json** files containing the object-wise semantic information (**OSI**) annotations (i.e., object bounding box, object ID, class, status, position, direction, distance, and velocity) made in a total of 2532 objects over 309 images from the validation set of the proposed MOD dataset. Each **.json** file contains the annotations for the corresponding image file.

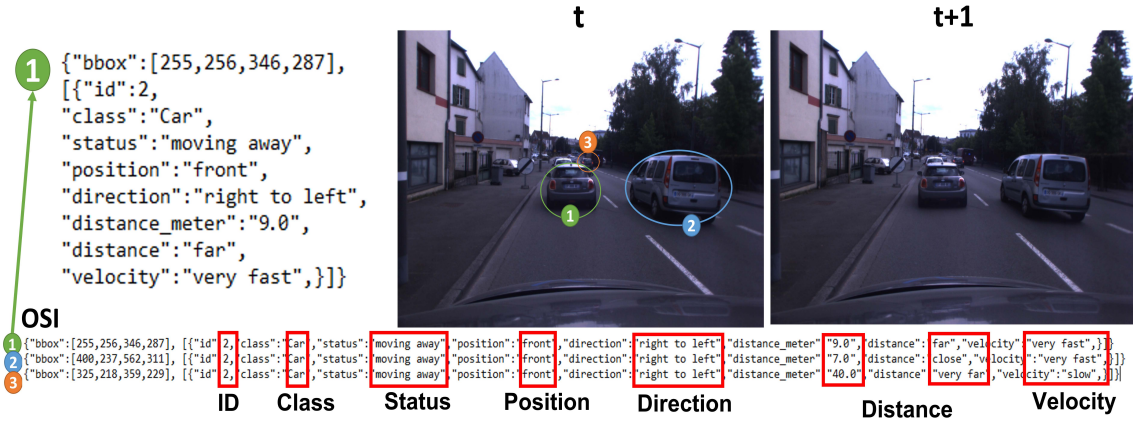


Figure 4.15: Manually annotated Object-wise Semantic Information (**OSI**) from two consecutive images (t and $t+1$).

$$
\begin{aligned}
&annotation\{\\
&\quad "object_{bbox}" : [x, y, width, height],\\
&\quad "object_{attributes}" : str,\\
&\}\\
&object_{attributes}[\{\\
&\quad "id" : int,\\
&\quad "class" : chr,\\
&\quad "status" : chr,\\
&\quad "position" : chr,\\
&\quad "direction" : chr,\\
&\quad "distance_{meter}" : float,\\
&\quad "distance" : chr,\\
&\quad "velocity" : chr,\\
&\}]
\end{aligned}
$$

For each object, a new line is created. 4.15 is an example of annotation format where the image contains several objects.

The evaluation is based on the matching of the predicted (Pred) information extracted by FOI (read from $Pred_{json}$ file) against the ground truth (GT) **OSI** (read from GT_{json} file). Figure 4.16 defines the structure of GT and Pred OSIs.

4.4.1/ EVALUATION - PART I

We calculate the accuracy of each attribute individually i.e., object class, status (moving/static), object motion (approaching, away, left-to-right, right-to-left), position (front,front-left,-right), and distance (very close, close, far, very far) using precision metric.

$$Accuracy_{obj}^{Attribute} = \frac{Attribute_{Correct}}{(Attribute_{Correct} + Attribute_{False})} \tag{4.15}$$

The total accuracy of the task is calculated by

$$Accuracy_{overall}^{Attribute} = \frac{\Sigma \quad Attribute_{Correct}}{\Sigma \quad (Attribute_{Correct} + Attribute_{False})} \tag{4.16}$$

Attributes:

$Pred_{LINE} = [Object_{ID}, Object_{Class}, Object_{Status}, Object_{Position}, Object_{Direction}, Object_{Distance}, Object_{Velocity}]$

$GT_{LINE} = [Object_{ID}, Object_{Class}, Object_{Status}, Object_{Position}, Object_{Direction}, Object_{Distance}, Object_{Velocity}]$

OBJECT CLASS

The class accuracy of each object is given by

$$Accuracy_{object}^{class} = \frac{Class_{Correct}}{(Class_{Correct} + Class_{False})} \tag{4.17}$$

if $Pred_{LINE}[1] == GT_{LINE}[1]$ **then**
  $Object_{Class} = Correct$

else
  $Object_{Class} = False$

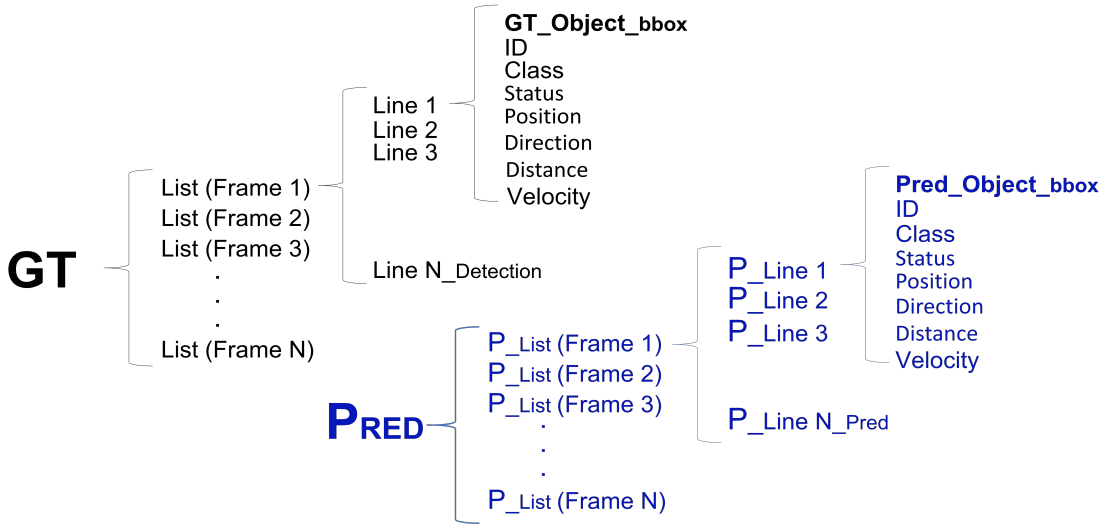


Figure 4.16: Example **OSI** Tree (Ground Truth and Predicted)

OBJECT STATUS

The accuracy of each moving and static object is given by

$$Accuracy_{object}^{Moving} = \frac{Moving_{Correct}}{(Moving_{Correct} + Moving_{False})} \tag{4.18}$$

$$Accuracy_{object}^{Static} = \frac{Static_{Correct}}{(Static_{Correct} + Static_{False})} \tag{4.19}$$

if $Pred_{LINE}[2]$ == "moving" & $GT_{LINE}[2]$== "moving" **then**
$Object_{Status} = Moving_{Correct}$

if $Pred_{LINE}[2]$ == "static" & $GT_{LINE}[2]$== "moving" **then**
$Object_{Status} = Moving_{False}$

if $Pred_{LINE}[2]$ == "moving" & $GT_{LINE}[2]$== "static" **then**
$Object_{Status} = Static_{False}$

if $Pred_{LINE}[2]$ == "static" & $GT_{LINE}[2]$== "static" **then**
$Object_{Status} = Static_{Correct}$

OBJECT MOTION

The motion accuracy includes position, direction,and velocity are given by

Position :

$$Accuracy_{object}^{Position} = \frac{Position_{Correct}}{(Position_{Correct} + Position_{False})} \tag{4.20}$$

if $Pred_{LINE}[3]$ == $GT_{LINE}[3]$ **then**
$Object_{Position} = Correct$

else
$Object_{Position} = False$

Direction :

$$Accuracy_{object}^{Direction_{Approaching}} = \frac{Approaching_{Correct}}{(Approaching_{Correct} + Approaching_{False})} \tag{4.21}$$

if $Pred_{LINE}[4]$ = "approaching" & $GT_{LINE}[4]$= "approaching" **then**
$Object_{Direction} = Approaching_{Correct}$

if $Pred_{LINE}[4]$ = "moving away" or "motionless" or "L2R" or "R2L" & $GT_{LINE}[4]$= "approaching" **then**
$Object_{Direction} = Approaching_{False}$

$$Accuracy_{object}^{Direction_{Away}} = \frac{Away_{Correct}}{(Away_{Correct} + Away_{False})} \tag{4.22}$$

if $Pred_{LINE}[4]$ = "moving away" & $GT_{LINE}[4]$= "moving away" **then**
$Object_{Direction} = Away_{Correct}$

if $Pred_{LINE}[4]$ = "approaching" or "motionless" or "L2R" or "R2L" & $GT_{LINE}[4]$= "moving away" **then**
$Object_{Direction} = Away_{False}$

$$Accuracy_{object}^{Direction_{L2R}} = \frac{L2R_{Correct}}{(L2R_{Correct} + L2R_{False})} \tag{4.23}$$

if $Pred_{LINE}[4]$ = "L2R" & $GT_{LINE}[4]$== "L2R" **then**
$Object_{Direction} = L2R_{Correct}$

if $Pred_{LINE}[4]$ = "moving away" or "motionless" or "approaching" or "R2L" & $GT_{LINE}[4]$= "L2R" **then**
$Object_{Direction} = L2R_{False}$

$$Accuracy_{object}^{Direction_{R2L}} = \frac{R2L_{Correct}}{(R2L_{Correct} + R2L_{False})} \tag{4.24}$$

if $Pred_{LINE}[4]$ = "R2L" & $GT_{LINE}[4]$= "R2L" **then**
$Object_{Direction} = R2L_{Correct}$

if $Pred_{LINE}[4]$ = "moving away" or "motionless" or "approaching" or "L2R" & $GT_{LINE}[4]$= "R2L" **then**
$Object_{Direction} = R2L_{False}$

Velocity :

$$Accuracy_{object}^{Velocity_{Slow}} = \frac{Slow_{Correct}}{(Slow_{Correct} + Slow_{False})} \tag{4.25}$$

if $Pred_{LINE}[6]$ == "slow" & $GT_{LINE}[6]$== "slow" **then**
$Object_{Velocity} = Slow_{Correct}$

if $Pred_{LINE}[6]$ == "medium" or "fast" or "very fast" & $GT_{LINE}[6]$== "slow" **then**
$Object_{Velocity} = Slow_{False}$

$$Accuracy_{object}^{Velocity_{Medium}} = \frac{Medium_{Correct}}{(Medium_{Correct} + Medium_{False})} \tag{4.26}$$

if $Pred_{LINE}[6]$ == "medium" & $GT_{LINE}[6]$== "medium" **then**
$Object_{Velocity} = Medium_{Correct}$

if $Pred_{LINE}[6]$ == "slow" or "fast" or "very fast" & $GT_{LINE}[6]$== "medium" **then**
$Object_{Velocity} = Medium_{False}$

$$Accuracy_{object}^{Velocity_{Fast}} = \frac{Fast_{Correct}}{(Fast_{Correct} + Fast_{False})} \tag{4.27}$$

if $Pred_{LINE}[6]$ == "fast" & $GT_{LINE}[6]$== "fast" **then**
$Object_{Velocity} = Fast_{Correct}$

if $Pred_{LINE}[6]$ == "slow" or "medium" or "very fast" & $GT_{LINE}[6]$== "fast" **then**
$Object_{Velocity} = Fast_{False}$

$$Accuracy_{object}^{Velocity_{VFast}} = \frac{VFast_{Correct}}{(VFast_{Correct} + VFast_{False})} \tag{4.28}$$

if $Pred_{LINE}[6]$ == "very fast" & $GT_{LINE}[6]$== "very fast" **then**
$Object_{Velocity} = VFast_{Correct}$

if $Pred_{LINE}[6]$ == "slow" or "medium" or "fast" & $GT_{LINE}[6]$== "very fast" **then**
$Object_{Velocity} = VFast_{False}$

OBJECT DISTANCE

The accuracy of distance is given by

$$Accuracy_{object}^{Distance_{VClose}} = \frac{VClose_{Correct}}{(VClose_{Correct} + VClose_{False})} \tag{4.29}$$

if $Pred_{LINE}[5]$ = "very close" & $GT_{LINE}[5]$== "very close" **then**
$Object_{Distance} = VClose_{Correct}$

if $Pred_{LINE}[5]$ = "close" or "far" or "very far" & $GT_{LINE}[5]$== "very close" **then**
$Object_{Distance} = VClose_{False}$

$$Accuracy_{object}^{Distance_{Close}} = \frac{Close_{Correct}}{(Close_{Correct} + Close_{False})} \tag{4.30}$$

if $Pred_{LINE}[5]$ = "close" & $GT_{LINE}[5]$== "close" **then**
$Object_{Distance} = Close_{Correct}$

if $Pred_{LINE}[5]$ = "very close" or "far" or "very far" & $GT_{LINE}[5]$== "close" **then**
$Object_{Distance} = Close_{False}$

$$Accuracy_{object}^{Distance_{Far}} = \frac{Far_{Correct}}{(Far_{Correct} + Far_{False})} \tag{4.31}$$

if $Pred_{LINE}[5]$ = "far" & $GT_{LINE}[5]$== "far" **then**
$Object_{Distance} = Far_{Correct}$

if $Pred_{LINE}[5]$ = "very close" or "close" or "very far" & $GT_{LINE}[5]$== "far" **then**
$Object_{Distance} = Far_{False}$

$$Accuracy_{object}^{Distance_{VFar}} = \frac{VFar_{Correct}}{(VFar_{Correct} + VFar_{False})} \tag{4.32}$$

if $Pred_{LINE}[5]$ = "very far" & $GT_{LINE}[5]$== "very far" **then**
$Object_{Distance} = VFar_{Correct}$

if $Pred_{LINE}[5]$ = "very close" or "close" or "far" & $GT_{LINE}[5]$== "very far" **then**
  $Object_{Distance} = VFar_{False}$

We computed the overall accuracy of the moving/static, motion, velocity, and distance of the detected objects (using equation 4.16) over 309 images from the validation set of the proposed MOD dataset. **Tables 4.7**, and **4.8** shows the predicted accuracy scores.

Table 4.7: Accuracies for Moving and Static Objects

No of frames	No of Objects	Moving Objects	Static Objects	Over-All Accuracy (%)	
				Moving Object Identification	Static Object Identification
309	2532	723	1809	**75.55**	**94.33**

Table 4.8: Accuracies for Movement, Distance, Velocity and Position

No of frames	Movement Accuracy (%)				
	Apr	Awy	R2L	L2R	**Over-All**
	50.61	87.92	83.01	86.03	**87.48**
	Distance Accuracy (%)				
	vFar	Far	Close	vClose	**Over-All**
	90.13	93.34	98.73	95.40	**94.69**
309	**Velocity Accuracy (%)**				
	Slow	Mdm	Fast	vFast	**Over-All**
	74.74	80.35	90.30	95.00	**83.28**
	Position Accuracy (%)				
	Frnt L	Front		Frnt R	**Over-All**
	100.00	100.00		100.00	**100.00**

4.4.2/ EVALUATION - PART II

In the second part, we evaluate the overall accuracy of FOI, which depends on the correct extraction of each OSI, which we call object identification accuracy. For example, all the information about the $Object_{Class}$, $Object_{Moving}$, $Object_{Static}$, $Object_{Position}$, $Object_{distance}$, $Object_{motion}$, and $Object_{velocity}$ must be predicted **CORRECTLY**. If any of the predictions are wrong, the system should consider it a **FALSE** or incorrect identification. We also calculate the computation time of the FOI. **Table 4.9** illustrates the overall object identification accuracy of the FOI, and **Table 4.10** shows the performance (processing time) of each task within the FOI and the overall speed.

Table 4.9: Over-All Accuracy of FOI

No of Frames	No of Objects		FOI Over-All Accuracy (%)
	Moving	Static	
309	723	1809	**81.27**

COMPUTATIONAL TIME

All experiments were performed on a standard desktop (Intel core i9, two RTX 2080Ti GPUs, using 375×1242 input images) with the Python processing environment. The average computation time for motion estimation/compensation is about 0.1153 seconds (image registration takes 0.0652 seconds and optical flow takes 0.0501 seconds) for each frame. The moving object detection model takes about 0.1045 seconds, and the disparity map calculation step takes about 0.0438 seconds per frame. About 0.0094 seconds per image is required for fusion and information extraction. The total inference time of the proposed FOI is about 0.1247 seconds per frame, which is equivalent to **8.02 fps**.

Table 4.10: Performance (processing time) within FOI and overall speed using 375×1242 input images

Task	Inference Time [s/f]
Motion Estimation / Compensation	0.1153
MOD	0.1045
Disparity	0.0438
Information Fusion	0.0094
FOI	0.1247

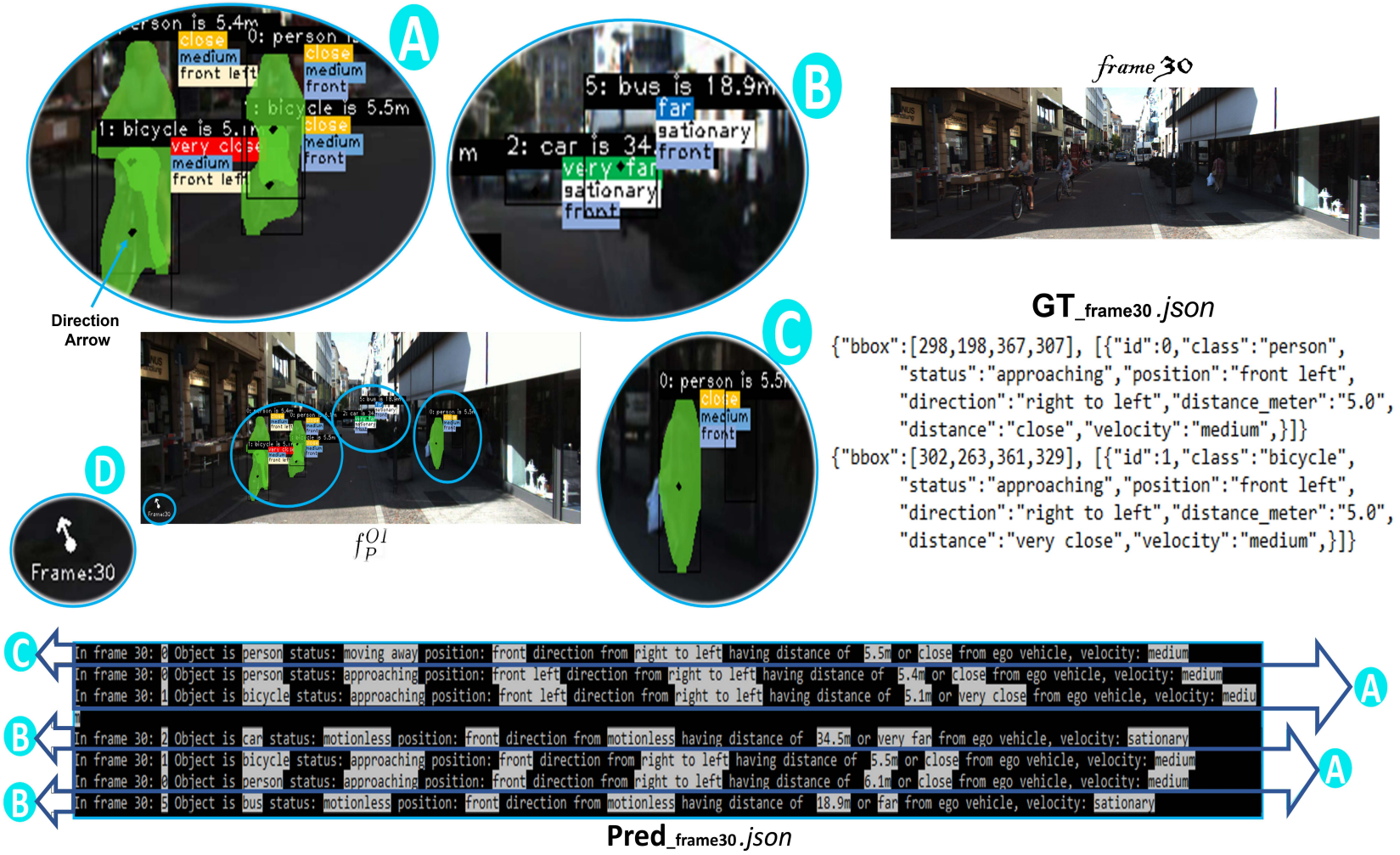


Figure 4.17: Example predicted object identification f_P^{OI} (Highlighting the labeling and colorized scaling) of sample frame 30. The frame work generates JSON file of the predicted OSIs ($Pred_{frame30}.json$) and later compared with the Ground truth OSI ($GT_{frame30}.json$) for the evaluation.

Figure 4.17 shows an example of the predicted output f_P^{OI} and json file $json_{file}$ generated by the proposed FOI for an input frame f_{30}.

A total of seven objects are detected, five of which are moving and two of which are static. The moving objects are labeled/segmented and colored green, while the static objects are only labeled with a bounding box. Each object is shown with the labels ID /class and the distance in meters above the bounding box. The black arrow in the center of the bounding box indicates the direction of the object and marks it in the generated output json file $json_{file}$ as approaching, moving away, right to left, and left to right. The distance, position, and velocity of each object are color-coded according to scale and displayed in the bounding box of the corresponding object in the upper right corner. In the city scenario example f_P^{OI}, different object types occur 4.17, e.g., a car, a pedestrian, two cyclists.

Zoom (A) shows that there are two moving cyclists, and both of them approach the ego vehicle. FOI successfully identifies their direction of movement, position with respect to the ego-vehicle, velocity, and distance from the ego-vehicle.

In Zoom (C), a pedestrian is detected in $front$ having a distance of $5.5m$, who is $moving - away$ from the ego-vehicle with $medium$ velocity.

Zoom (B) shows a car and a bus (actually a van) as static and stationary objects, respectively. The detected car is motionless and very far away from the ego-vehicle, while the bus is parked on the side of the road. The exact identification of these objects by the FOI can be seen in the json file output *"Object is car status: motionless position: front direction from motionless having a distance of 34.5m or very far from ego vehicle, velocity stationery"* can be seen.

The white arrow in the lower left corner shows the direction of movement of the ego-vehicle and the frame number (shown in zoom (D)).

The qualitative results of FOI on different sequence runs are shown in **Figures 4.18**, **4.19**, and **4.20**, respectively.

Figure 4.18: FOI Results on different sequence runs.

Figure 4.19: FOI Results on different sequence runs.

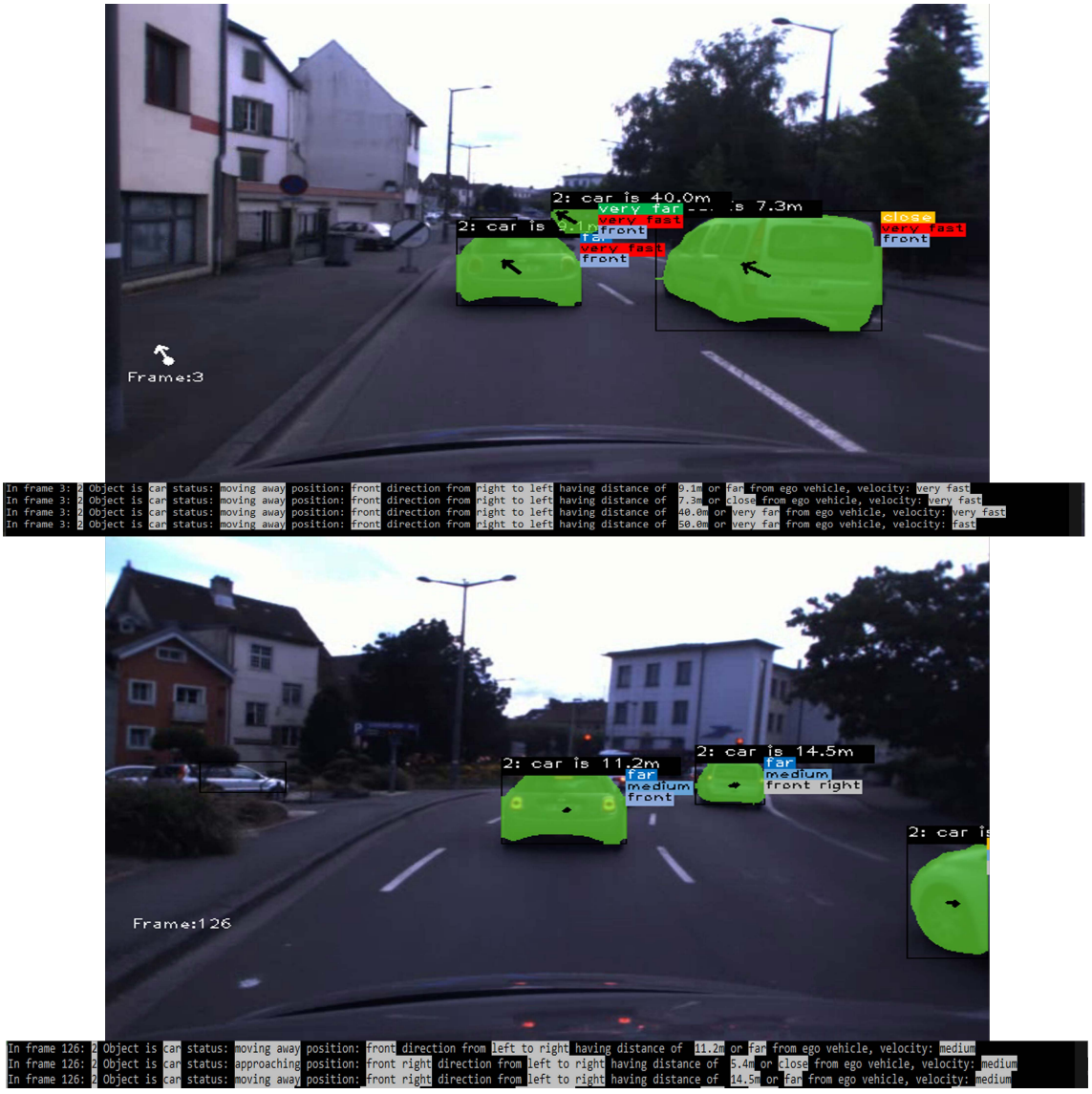


Figure 4.20: FOI Results on different sequence runs.

The **figures 4.18, 4.19 and 4.20** show ten output examples of our framework for object identification in urban driving situations. In these scenarios, different types of moving objects occur, such as cars, motorcycles, and bicycles. In the second image of the figure 4.18, seven objects are detected. Among them, three are moving objects (car, person, bicycle), and four (two cars, two bicycles) are static. The OSI of each object is labeled and highlighted. E.g., object type ”car” with status ”moving away” from the ego-vehicle, position ”front,” and the direction of movement is ”left to right” with a distance of ”7.2m” from the ego-vehicle, which is scaled as ”close” to the ego-vehicle. The two bicycles standing on the sidewalk are correctly detected by FOI, with status ”stationary,” direction ”motionless,” position ”front right,” and distance ”close” to the ego-vehicle. The person on the bike ”in front” is ”moving away,” ”fast” from ”right to left,” and ”very close” to the ego-vehicle. Image three of the figure 4.18 shows a vehicle ”in front” moving ”away” from ego-

vehicle at a ”slow” velocity. Person ”far” moving ”slowly” from ”left to right” and three static vehicles, one in front and two in front right of ego- vehicle. A motorcycle approaching fast can be seen in the first image of the figure 4.19, along with static cars parked on the side of the road very far from the ego-vehicle. Image three of figure 4.19 shows several pedestrians moving in different directions and taking different positions. In figure 4.20 image two, the three cars in front are moving very fast. Detailed results containing consecutive frames of our proposed FOI could be found at https://youtube/.... The results show that our proposed FOI achieves promising results in identifying the objects within the driving scene. The objects include vehicles, buses, trucks, trains, pedestrians, cyclists and motorcyclists.

The performance of the proposed FOI could be affected by several typical factors, including object motion speed, ego-vehicle speed, overlapping or very close objects, object reflectance, and object size. It was found that when the velocities of the moving object and the ego-vehicle are the same (for the same direction) or cancel (for the opposite direction), the system detects the object as static. e.g., in column two image 3 of Figure 4.21, the blue vehicle approaches the ego-vehicle and is detected as static. A similar thing happens in column two image 4 of Figure 4.21, the red car moving in front of the ego vehicle is detected as static. The problem also occurs when the object is far away from the ego-vehicle and moving very slowly. The images in column one of Figure 4.21

Figure 4.21: FOI False Detections, Affected by object motion speed/ego-vehicle speed, objects overlap, object reflection etc.

show false detection due to object overlap, and images 1 and 2 in column one show an example of object reflection. However, such false detections and wrong classifications are very rare in our proposed framework.

4.5/ CONCLUSION

A new framework has been proposed to identify the objects of a moving camera in a complex driving scene using various image processing techniques. The system focuses on detecting objects and extracting their characteristics in terms of motion, position, distance and velocity. In addition, we have addressed the problem of extracting the actual motion information of moving objects from a moving camera using image registration and optical flow estimation. A deep neural network (MOD) moving object detection model based on combining a segmentation network with an encoder-decoder network was proposed, which can detect the pixel-wise motion state (moving/static) from two consecutive images. In addition, a new dataset for moving object detection has been proposed, which includes all types of vehicles, pedestrians, cyclists and motorcyclists. Evaluation of MOD has been performed using both the proposed dataset and the existing MOD dataset. We have shown that the performance of the proposed MOD model outperforms state-of-the-art MOD methods in terms of accuracy while providing competitive time inference. We obtained the best results in the KITTI-Motion, KITTI-MoSeg and KITTI-MoSeg Extended datasets. Fully motion compensated optical flow maps are obtained by combining the results from MOD and the registered optical flow. The information related to object class, status, motion, position, velocity and distance are extracted from MOD, compensated optical flow maps and disparity maps. All these pieces of information or object-wise semantic information (OSI) are highlighted as colored labels on the bounding boxes of each object. The experimental results in different sequences show that the proposed framework is robust in terms of camera movement and correct object identification. This information would help to plan the ego-trajectories based on the future states of the identified objects, thus avoiding collision risks and assisting ADAS in decision making. Except for the camera sensor, our approach does not rely on data from active sensors. Overall, FOI provides a high accuracy of **81.27%** and an acceptable processing rate of **8.02 fps** in multiple sequences. An important issue is the computational complexity of the proposed framework. This is mainly due to the computation of image registration and optical flow, as their estimation is often difficult and time-consuming in a complex dynamic environment. GPU-based techniques could be used to overcome this weakness.

III

CONCLUSION

5

GENERAL CONCLUSION

This chapter summarizes the contributions of the thesis with the main conclusions and recommendations for future research.

5.1/ SUMMARY OF THE PHD THESIS

In this thesis, we focused on the problem of visual scene understanding by recognizing the semantic constituents of a driving scene. The underlying theme of this thesis was to investigate, design, implement and evaluate the Deep Learning based solutions for semantic analysis of the driving environment in urban scenarios. We have proposed several novel Deep Learning methods for visual scene understanding using only image data. We described several theoretical contributions for the proposed methods, reported qualitative and quantitative results through extensive experimental evaluation on standard benchmarks and in different real-world environments, discussed related work, and demonstrated that our proposed architectures substantially exceed the state-of-the-art.

At the beginning, we gave a comprehensive overview of deep-learning-based methods for semantic segmentation, a very well- studied topic and one of the fundamental problems in scene understanding. Existing deep methods are grouped according to a common taxonomy: Concept (fully convolutional, encoder-decoder architectures, multiscale and pyramid-based approaches, Atrous/Dilated convolutional models, recurrent networks, regional proposal-based methods, Transformers, generative models in adversarial setting, context-aware models, semi-supervised and weakly supervised methods), network architecture (highlighting their contributions to model design), architecture origin (inspire or deviate from previous SOTA methods), test benchmarks, and code availability. A detailed review of publicly available benchmark datasets is presented, including data type/nature, number of classes, image resolution, year of publication, and peak performance achieved by the network model (up to the submission of this thesis). Furthermore, we described the common evaluation metrics for semantic segmentation. In addition, we explored the

similarity, strengths, and challenges of the deep learning models based on their design strategies and evaluated performance. Moreover, we discussed some open problems and their possible solutions for deep learning-based semantic segmentation. The aim of this study was to provide the reader with a comprehensive and heuristic overview of deep learning based semantic segmentation techniques. The comprehensive description of network architecture design and datasets can help new researchers to strengthen their understanding, make comparisons or select methods and datasets according to their application and requirements.

The second part of the thesis deals with an autonomous driving system that is able to pay attention to the most important objects/regions in the driving environment. We proposed a novel idea for visual attention for driving images that highlight objects in the road context as salient using a Generative Adversarial Network. We first investigated well-known saliency algorithms, including classical and deep learning approaches, and their applicability to visual saliency for multiple objects in driving scenes. We concluded that none of these tested algorithms could work in complex and diverse environments such as driving. We developed a new strategy for data generation and visual saliency prediction. We investigated the robustness of different algorithms for computing visual saliency for images corrupted by white Gaussian noise, and concluded that the VSF algorithm is best suited to construct the ground truth for our proposed attention system. Data are collected from publicly available driving datasets Yu et al. (2018), Fauqueur et al. (2007), Cordts et al. (2016) that include various driving activities and environments, including rain, night, snow, highways, and urban scenes. Experimental results and quantitative and qualitative comparisons with SOTA saliency and eye fixation attention models demonstrate the ability of our system to predict several important objects in interactive, complex, and dynamic driving environments.

The environment in which an autonomous vehicle moves evolves regularly, and this evolution is closely related to its motion and semantic characteristics. In the third part of the thesis, we focused on the motion characteristics of objects in urban driving scenarios. To this end, we have proposed a framework for object identification that focuses on detecting objects from a moving camera and extracting their characteristics such as object type/class, status (moving/static), direction, distance and position to the ego-vehicle, and object velocity. The proposed work is the first approach that uses image registration along with optical flow estimation to extract the actual motion of moving objects from the moving camera. We introduced a deep neural network (MOD) moving object detection model based on the combination of a segmentation network with an encoder-decoder network, which can detect the pixel-wise motion state of the object (moving/static) from two consecutive images. The recognition of objects in the framework is not object type specific. We created a new dataset for moving object detection that includes all vehicles, pedestrians, cyclists, and motorcyclists. We reported the superior performance in

terms of accuracy of our proposed MOD technique compared to state-of-the-art methods on a publicly available MOD dataset (KITTI-Motion, KITTI-MoSeg, and KITTI-MoSeg Extended). We incorporated the well-known SGM algorithm for disparity estimation. We advanced by combining the outcomes of actual motion estimation and moving object detection network to fully compensate the camera motion. The information related to object class, status, motion, position, velocity and distance are extracted from MOD, compensated optical flow maps and disparity maps. All these pieces of information or object-wise semantic information (OSI) are highlighted as colored labels on the bounding boxes of each object, and also the OSI of each object is stored in a $json$ file. The final evaluation is based on matching the predicted OSIs (from $json$) against manually annotated ground truth OSIs (from $json$). The experimental results in several different sequences show that the proposed framework is robust in terms of camera motion and correct object identification. This work aimed to combine the motion and semantic characteristics of objects in the urban driving environment using image processing-based techniques. The result information from the proposed framework can help ADAS or autonomous vehicles in situation interpretation (with prior knowledge such as traffic rules and knowledge from previous experiences), identify potential threats, provide more accurate warnings to a human driver, and data to an intelligent agent module responsible for decision making.

5.2/ FUTURE PERSPECTIVES

In the field of autonomous driving, there are many technical challenges that still leave much room for development. These challenges are related to sensors, computer hardware, mapping and localization, planning, decision making, and control. Our work does not address all challenges, but is limited to the part of perception in driving, including semantic segmentation, visual attention, moving object detection, motion compensation/estimation, and disparity estimation, which plays a crucial role in planning and decision making.

The perspectives of this work include, first, the development of a new network model for semantic segmentation to improve the accuracy and computational efficiency of segmentation networks for autonomous driving applications. Our study (survey) has shown that CNN-based semantic segmentation approaches suffer from higher-order inconsistencies between the ground-truth labels and the labels predicted by the segmentation model. In our approach, adversarial learning (Generative Adversarial Network) is used as a post-processing method to make the semantic segmentation network output more realistic, refined and better structure-preserving (closer to ground-truth).

Second, the perceptual data (object identification) can be used to plan safe and smooth trajectories for the objects of interest, taking into account their dynamic limits, navigational

convenience and safety, and traffic rules. One can further extend our work on object identification and add other sources of information, such as object tracking, line detection, traffic signs, and live traffic light detection, to determine the relevance of objects depending on the driving situation. Adding this additional information to the proposed framework could help prioritize the detected objects and help in various tasks such as lane change, obstacle avoidance, highlighting detected objects in different priority levels (critical, high, medium, and low), and handling critical driving situations.

Third, the deployment of the our object identification framework on a autonomous vehicles to validate this work and define the limitations and challenges in a real scenario. The biggest challenge in validating autonomous vehicles is safety (protection of road users). Such utility is very important and therefore requires the system to be robust and reliable. Our validation process must test whether the system can detect object attributes precisely, and whether it can function successfully in bad weather or adverse environmental conditions.

From this study, it is inferred that the existing optical flow dataset for driving scenarios, where the images are captured by a moving camera does not reflect the true distribution of the apparent motion velocities of the brightness pattern. We believe that an accurate optical flow dataset or a compensated optical flow dataset is needed that assigns an accurate and precise color to each vector based on its orientation. In the future, we will attempt to use image registration to create new ground truths for optical flow and create a new dataset.

BIBLIOGRAPHY

[Achanta et al. 2012] ACHANTA, Radhakrishna ; SHAJI, Appu ; SMITH, Kevin ; LUCCHI, Aurelien ; FUA, Pascal ; SÜSSTRUNK, Sabine ; OTHERS: **"SLIC superpixels compared to state-of-the-art superpixel methods"**. In *IEEE transactions on pattern analysis and machine intelligence* 34 (2012), number 11, pages 2274–2282

[Adams et al. 2010a] ADAMS, Andrew ; BAEK, Jongmin ; DAVIS, Myers A.: **"Fast high-dimensional filtering using the permutohedral lattice"**. In *Computer Graphics Forum* Volume 29 Wiley Online Library (event), 2010, pages 753–762

[Adams et al. 2010b] ADAMS, Andrew ; BAEK, Jongmin ; DAVIS, Myers A.: **"Fast high-dimensional filtering using the permutohedral lattice"**. In *Computer Graphics Forum* Volume 29 Wiley Online Library (event), 2010, pages 753–762

[Alletto et al. 2016] ALLETTO, Stefano ; PALAZZI, Andrea ; SOLERA, Francesco ; CALDERARA, Simone ; CUCCHIARA, Rita: **"Dr (eye) ve: a dataset for attention-based tasks with applications to autonomous and assisted driving"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pages 54–60

[Alvarez et al. 2012] ALVAREZ, Jose M. ; LECUN, Yann ; GEVERS, Theo ; LOPEZ, Antonio M.: **"Semantic road segmentation via multi-scale ensembles of learned features"**. In *European Conference on Computer Vision* Springer (event), 2012, pages 586–595

[Amirul Islam et al. 2017] AMIRUL ISLAM, Md ; ROCHAN, Mrigank ; BRUCE, Neil D. ; WANG, Yang: **"Gated feedback refinement network for dense image labeling"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pages 3751–3759

[Armeni et al. 2017] ARMENI, Iro ; SAX, Sasha ; ZAMIR, Amir R. ; SAVARESE, Silvio: **"Joint 2d-3d-semantic data for indoor scene understanding"**. In *arXiv preprint arXiv:1702.01105* (2017)

[Arnab et al. 2016] ARNAB, Anurag ; JAYASUMANA, Sadeep ; ZHENG, Shuai ; TORR, Philip H.: **"Higher order conditional random fields in deep neural networks"**. In *European Conference on Computer Vision* Springer (event), 2016, pages 524–540

[Arnab et al. 2018] ARNAB, Anurag ; ZHENG, Shuai ; JAYASUMANA, Sadeep ; ROMERA-PAREDES, Bernardino ; LARSSON, Mns ; KIRILLOV, Alexander ; SAVCHYNSKYY, Bogdan ; ROTHER, Carsten ; KAHL, Fredrik ; TORR, Philip: **"Conditional random fields meet deep neural networks for semantic segmentation"**. In *IEEE Signal Processing Magazine* 2 (2018)

[Awad et al. 2018] AWAD, Edmond ; DSOUZA, Sohan ; KIM, Richard ; SCHULZ, Jonathan ; HENRICH, Joseph ; SHARIFF, Azim ; BONNEFON, Jean-François ; RAHWAN, Iyad: **"The moral machine experiment"**. In *Nature* 563 (2018), number 7729, pages 59

[Badrinarayanan et al. 2017] BADRINARAYANAN, Vijay ; KENDALL, Alex ; CIPOLLA, Roberto: **"Segnet: A deep convolutional encoder-decoder architecture for image segmentation"**. In *IEEE transactions on pattern analysis and machine intelligence* 39 (2017), number 12, pages 2481–2495

[Barnes et al. 2017] BARNES, Dan ; MADDERN, Will ; POSNER, Ingmar: **"Find your own way: Weakly-supervised segmentation of path proposals for urban autonomy"**. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on* IEEE (event), 2017, pages 203–210

[Barron and Poole 2016] BARRON, Jonathan T. ; POOLE, Ben: **"The fast bilateral solver"**. In *European Conference on Computer Vision* Springer (event), 2016, pages 617–632

[Bazzani et al. 2016] BAZZANI, Loris ; LAROCHELLE, Hugo ; TORRESANI, Lorenzo: **"Recurrent mixture density network for spatiotemporal visual attention"**. In *arXiv preprint arXiv:1603.08199* (2016)

[Bertasius et al. 2016] BERTASIUS, Gedas ; SHI, Jianbo ; TORRESANI, Lorenzo: **"Semantic segmentation with boundary neural fields"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pages 3602–3610

[Bertasius et al. 2017] BERTASIUS, Gedas ; TORRESANI, Lorenzo ; STELLA, X Y. ; SHI, Jianbo: **"Convolutional Random Walk Networks for Semantic Image Segmentation."**. In *CVPR*, 2017, pages 6137–6145

[Bilinski and Prisacariu 2018] BILINSKI, Piotr ; PRISACARIU, Victor: **"Dense decoder shortcut connections for single-pass semantic segmentation"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pages 6596–6605

[Bloomfield and Steiger 1983] BLOOMFIELD, Peter ; STEIGER, William L.: *Least absolute deviations: theory, applications, and algorithms*. Springer, 1983

[Borghi et al. 2017] BORGHI, Guido ; VENTURELLI, Marco ; VEZZANI, Roberto ; CUCCHIARA, Rita: **"Poseidon: Face-from-depth for driver pose estimation"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pages 4661–4670

[Borji 2019] BORJI, Ali: **"Saliency Prediction in the Deep Learning Era: Successes and Limitations"**. In *IEEE transactions on pattern analysis and machine intelligence* (2019)

[Boyd and Vandenberghe 2004] BOYD, Stephen ; VANDENBERGHE, Lieven: *Convex optimization*. Cambridge university press, 2004

[Brostow et al. 2009] BROSTOW, Gabriel J. ; FAUQUEUR, Julien ; CIPOLLA, Roberto: **"Semantic object classes in video: A high-definition ground truth database"**. In *Pattern Recognition Letters* 30 (2009), number 2, pages 88–97

[Bulo et al. 2018] BULO, Samuel R. ; PORZI, Lorenzo ; KONTSCHIEDER, Peter: **"In-place activated batchnorm for memory-optimized training of dnns"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pages 5639–5647

[Byeon et al. 2015] BYEON, Wonmin ; BREUEL, Thomas M. ; RAUE, Federico ; LIWICKI, Marcus: **"Scene labeling with lstm recurrent neural networks"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pages 3547–3555

[Bylinskii et al. 2016] BYLINSKII, Zoya ; JUDD, Tilke ; OLIVA, Aude ; TORRALBA, Antonio ; DURAND, Frédo: **"What do different evaluation metrics tell us about saliency models?"**. In *arXiv preprint arXiv:1604.03605* (2016)

[Bylinskii et al. 2018] BYLINSKII, Zoya ; JUDD, Tilke ; OLIVA, Aude ; TORRALBA, Antonio ; DURAND, Frédo: **"What do different evaluation metrics tell us about saliency models?"**. In *IEEE transactions on pattern analysis and machine intelligence* 41 (2018), number 3, pages 740–757

[Caesar et al. 2016] CAESAR, Holger ; UIJLINGS, Jasper ; FERRARI, Vittorio: **"COCO-Stuff: Thing and stuff classes in context"**. In *CoRR, abs/1612.03716* 5 (2016), pages 8

[Carion et al. 2020] CARION, Nicolas ; MASSA, Francisco ; SYNNAEVE, Gabriel ; USUNIER, Nicolas ; KIRILLOV, Alexander ; ZAGORUYKO, Sergey: **"End-to-end object detection with transformers"**. In *European Conference on Computer Vision* Springer (event), 2020, pages 213–229

[Casanova et al. 2018] CASANOVA, Arantxa ; CUCURULL, Guillem ; DROZDZAL, Michal ; ROMERO, Adriana ; BENGIO, Yoshua: **"On the iterative refinement of densely connected representation levels for semantic segmentation"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2018, pages 978–987

[Chandra and Kokkinos 2016] CHANDRA, Siddhartha ; KOKKINOS, Iasonas: **"Fast, exact and multi-scale inference for semantic image segmentation with deep gaussian crfs"**. In *European Conference on Computer Vision* Springer (event), 2016, pages 402–418

[Charles et al. 2017] CHARLES, R Q. ; SU, Hao ; KAICHUN, Mo ; GUIBAS, Leonidas J.: **"Pointnet: Deep learning on point sets for 3d classification and segmentation"**. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on* IEEE (event), 2017, pages 77–85

[Che et al. 2019] CHE, Zhaohui ; BORJI, Ali ; ZHAI, Guangtao ; MIN, Xiongkuo ; GUO, Guodong ; LE CALLET, Patrick: **"GazeGAN: A Generative Adversarial Saliency Model based on Invariance Analysis of Human Gaze During Scene Free Viewing"**. In *arXiv preprint arXiv:1905.06803* (2019)

[Chen et al. 2021a] CHEN, Jieneng ; LU, Yongyi ; YU, Qihang ; LUO, Xiangde ; ADELI, Ehsan ; WANG, Yan ; LU, Le ; YUILLE, Alan L. ; ZHOU, Yuyin: **"Transunet: Transformers make strong encoders for medical image segmentation"**. In *arXiv preprint arXiv:2102.04306* (2021)

[Chen et al. 2016a] CHEN, Liang-Chieh ; BARRON, Jonathan T. ; PAPANDREOU, George ; MURPHY, Kevin ; YUILLE, Alan L.: **"Semantic image segmentation with task-specific edge detection using cnns and a discriminatively trained domain transform"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pages 4545–4554

[Chen et al. 2014] CHEN, Liang-Chieh ; PAPANDREOU, George ; KOKKINOS, Iasonas ; MURPHY, Kevin ; YUILLE, Alan L.: **"Semantic image segmentation with deep convolutional nets and fully connected crfs"**. In *arXiv preprint arXiv:1412.7062* (2014)

[Chen et al. 2017a] CHEN, Liang-Chieh ; PAPANDREOU, George ; KOKKINOS, Iasonas ; MURPHY, Kevin ; YUILLE, Alan L.: **"Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs"**. In *IEEE transactions on pattern analysis and machine intelligence* 40 (2017), number 4, pages 834–848

[Chen et al. 2017b] CHEN, Liang-Chieh ; PAPANDREOU, George ; SCHROFF, Florian ;

ADAM, Hartwig: **"Rethinking atrous convolution for semantic image segmentation"**. In *arXiv preprint arXiv:1706.05587* (2017)

[Chen et al. 2016b] CHEN, Liang-Chieh ; YANG, Yi ; WANG, Jiang ; XU, Wei ; YUILLE, Alan L.: **"Attention to scale: Scale-aware semantic image segmentation"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pages 3640–3649

[Chen et al. 2016c] CHEN, Liang-Chieh ; YANG, Yi ; WANG, Jiang ; XU, Wei ; YUILLE, Alan L.: **"Attention to scale: Scale-aware semantic image segmentation"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pages 3640–3649

[Chen et al. 2018] CHEN, Liang-Chieh ; ZHU, Yukun ; PAPANDREOU, George ; SCHROFF, Florian ; ADAM, Hartwig: **"Encoder-decoder with atrous separable convolution for semantic image segmentation"**. In *Proceedings of the European conference on computer vision (ECCV)*, 2018, pages 801–818

[Chen et al. 2017c] CHEN, Long ; FAN, Lei ; XIE, Guodong ; HUANG, Kai ; NÜCHTER, Andreas: **"Moving-object detection from consecutive stereo pairs using slanted plane smoothing"**. In *IEEE Transactions on Intelligent Transportation Systems* 18 (2017), number 11, pages 3093–3102

[Chen et al. 2021b] CHEN, Xi ; HAN, Zhen ; LIU, Xiaoping ; LI, Zhiqiang ; FANG, Tao ; HUO, Hong ; LI, Qingli ; ZHU, Min ; LIU, Min ; YUAN, Haolei: **"Semantic boundary enhancement and position attention network with long-range dependency for semantic segmentation"**. In *Applied Soft Computing* (2021), pages 107511

[Cheng et al. 2021a] CHENG, Bowen ; SCHWING, Alex ; KIRILLOV, Alexander: **"Per-pixel classification is not all you need for semantic segmentation"**. In *Advances in Neural Information Processing Systems* 34 (2021)

[Cheng et al. 2021b] CHENG, Ho K. ; TAI, Yu-Wing ; TANG, Chi-Keung: **"Rethinking Space-Time Networks with Improved Memory Coverage for Efficient Video Object Segmentation"**. In *arXiv preprint arXiv:2106.05210* (2021)

[Cheng et al. 2021c] CHENG, Yuan ; YANG, Yuchao ; CHEN, Hai-Bao ; WONG, Ngai ; YU, Hao: **"S3-Net: A Fast and Lightweight Video Scene Understanding Network by Single-shot Segmentation"**. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2021, pages 3329–3337

[Cho et al. 2014] CHO, H. ; SEO, Y. ; KUMAR, B. V. K. V. ; RAJKUMAR, R. R.: **"A multi-sensor fusion system for moving object detection and tracking in urban driving environments"**. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pages 1836–1843. DOI: 10.1109/ICRA.2014.6907100

[Cho et al. 2014] CHO, Kyunghyun ; VAN MERRIËNBOER, Bart ; GULCEHRE, Caglar ; BAHDANAU, Dzmitry ; BOUGARES, Fethi ; SCHWENK, Holger ; BENGIO, Yoshua: **"Learning phrase representations using RNN encoder-decoder for statistical machine translation"**. In *arXiv preprint arXiv:1406.1078* (2014)

[Chollet 2017] CHOLLET, François: **"Xception: Deep learning with depthwise separable convolutions"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pages 1251–1258

[Cordts et al. 2016] CORDTS, Marius ; OMRAN, Mohamed ; RAMOS, Sebastian ; REHFELD, Timo ; ENZWEILER, Markus ; BENENSON, Rodrigo ; FRANKE, Uwe ; ROTH, Stefan ; SCHIELE, Bernt: **"The cityscapes dataset for semantic urban scene understanding"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pages 3213–3223

[Cordts et al. 2015] CORDTS, Marius ; OMRAN, Mohamed ; RAMOS, Sebastian ; SCHARWÄCHTER, Timo ; ENZWEILER, Markus ; BENENSON, Rodrigo ; FRANKE, Uwe ; ROTH, Stefan ; SCHIELE, Bernt: **"The cityscapes dataset"**. In *CVPR Workshop on the Future of Datasets in Vision* Volume 1, 2015, pages 3

[Cornia et al. 2016] CORNIA, Marcella ; BARALDI, Lorenzo ; SERRA, Giuseppe ; CUCCHIARA, Rita: **"A deep multi-level network for saliency prediction"**. In *2016 23rd International Conference on Pattern Recognition (ICPR)* IEEE (event), 2016, pages 3488–3493

[Cornia et al. 2018] CORNIA, Marcella ; BARALDI, Lorenzo ; SERRA, Giuseppe ; CUCCHIARA, Rita: **"Predicting human eye fixations via an lstm-based saliency attentive model"**. In *IEEE Transactions on Image Processing* 27 (2018), number 10, pages 5142–5154

[Dai et al. 2017] DAI, Angela ; CHANG, Angel X. ; SAVVA, Manolis ; HALBER, Maciej ; FUNKHOUSER, Thomas A. ; NIESSNER, Matthias: **"ScanNet: Richly-Annotated 3D Reconstructions of Indoor Scenes."**. In *CVPR* Volume 2, 2017, pages 10

[Dai et al. 2015] DAI, Jifeng ; HE, Kaiming ; SUN, Jian: **"Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation"**. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pages 1635–1643

[Deng et al. 2009] DENG, Jia ; DONG, Wei ; SOCHER, Richard ; LI, Li-Jia ; LI, Kai ; FEI-FEI, Li: **"Imagenet: A large-scale hierarchical image database"**. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on* Ieee (event), 2009, pages 248–255

[Deng et al. 2014] DENG, Tao ; CHEN, Andong ; GAO, Min ; YAN, Hongmei: **"Top-down based saliency model in traffic driving environment"**. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)* IEEE (event), 2014, pages 75–80

[Deng et al. 2017] DENG, Tao ; YAN, Hongmei ; LI, Yong-Jie: **"Learning to Boost Bottom-Up Fixation Prediction in Driving Environments via Random Forest"**. In *IEEE Transactions on Intelligent Transportation Systems* 19 (2017), number 9, pages 3059–3067

[Deng et al. 2020] DENG, Tao ; YAN, Hongmei ; QIN, Long ; NGO, Thuyen ; MANJUNATH, BS: **"How do drivers allocate their potential attention? Driving fixation prediction via convolutional neural networks"**. In *IEEE Transactions on Intelligent Transportation Systems* 21 (2020), number 5, pages 2146–2154

[Deng et al. 2016] DENG, Tao ; YANG, Kaifu ; LI, Yongjie ; YAN, Hongmei: **"Where does the driver look? Top-down-based saliency detection in a traffic driving environment"**. In *IEEE Transactions on Intelligent Transportation Systems* 17 (2016), number 7, pages 2051–2062

[Devlin et al. 2018] DEVLIN, Jacob ; CHANG, Ming-Wei ; LEE, Kenton ; TOUTANOVA, Kristina: **"Bert: Pre-training of deep bidirectional transformers for language understanding"**. In *arXiv preprint arXiv:1810.04805* (2018)

[Dickmanns 2002] DICKMANNS, Ernst D.: **"The development of machine vision for road vehicles in the last decade"**. In *Intelligent Vehicle Symposium, 2002. IEEE* Volume 1 IEEE (event), 2002, pages 268–281

[Ding et al. 2018] DING, Henghui ; JIANG, Xudong ; SHUAI, Bing ; LIU, Ai Q. ; WANG, Gang: **"Context contrasted feature and gated multi-scale aggregation for scene segmentation"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pages 2393–2402

[Dollar et al. 2011] DOLLAR, Piotr ; WOJEK, Christian ; SCHIELE, Bernt ; PERONA, Pietro: **"Pedestrian detection: An evaluation of the state of the art"**. In *IEEE transactions on pattern analysis and machine intelligence* 34 (2011), number 4, pages 743–761

[Dong et al. 2021] DONG, Zihao ; LI, Jinping ; FANG, Tiyu ; SHAO, Xiuli: **"Lightweight boundary refinement module based on point supervision for semantic segmentation"**. In *Image and Vision Computing* 110 (2021), pages 104169

[Dosovitskiy et al. 2020] DOSOVITSKIY, Alexey ; BEYER, Lucas ; KOLESNIKOV, Alexander ; WEISSENBORN, Dirk ; ZHAI, Xiaohua ; UNTERTHINER, Thomas ; DEHGHANI, Mostafa ; MINDERER, Matthias ; HEIGOLD, Georg ; GELLY, Sylvain ; OTHERS: **"An**

image is worth 16x16 words: Transformers for image recognition at scale". In *arXiv preprint arXiv:2010.11929* (2020)

[Duncan 1984] DUNCAN, John: **"Selective attention and the organization of visual information."**. In *Journal of Experimental Psychology: General* 113 (1984), number 4, pages 501

[Emre Yurdakul and Yemez 2017] EMRE YURDAKUL, Ekrem ; YEMEZ, Yucel: **"Semantic segmentation of rgbd videos with recurrent fully convolutional neural networks"**. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pages 367–374

[Everingham et al. 2015] EVERINGHAM, Mark ; ESLAMI, SM A. ; VAN GOOL, Luc ; WILLIAMS, Christopher K. ; WINN, John ; ZISSERMAN, Andrew: **"The pascal visual object classes challenge: A retrospective"**. In *International journal of computer vision* 111 (2015), number 1, pages 98–136

[Fan et al. 2018] FAN, Heng ; MEI, Xue ; PROKHOROV, Danil ; LING, Haibin: **"Multi-level contextual rnns with attention model for scene labeling"**. In *IEEE Transactions on Intelligent Transportation Systems* 19 (2018), number 11, pages 3475–3485

[Fang et al. 2019] FANG, Jianwu ; YAN, Dingxin ; QIAO, Jiahuan ; XUE, Jianru ; WANG, He ; LI, Sen: **"DADA-2000: Can Driving Accident be Predicted by Driver Attention? Analyzed by A Benchmark"**. In *arXiv preprint arXiv:1904.12634* (2019)

[Farabet et al. 2013] FARABET, Clement ; COUPRIE, Camille ; NAJMAN, Laurent ; LECUN, Yann: **"Learning hierarchical features for scene labeling"**. In *IEEE transactions on pattern analysis and machine intelligence* 35 (2013), number 8, pages 1915–1929

[Farnebäck 2003] FARNEBÄCK, Gunnar: **"Two-frame motion estimation based on polynomial expansion"**. In *Scandinavian conference on Image analysis* Springer (event), 2003, pages 363–370

[Fauqueur et al. 2007] FAUQUEUR, Julien ; BROSTOW, Gabriel ; CIPOLLA, Roberto: **"Assisted video object labeling by joint tracking of regions and keypoints"**. In *2007 IEEE 11th International Conference on Computer Vision* IEEE (event), 2007, pages 1–7

[Fayyaz et al. 2016] FAYYAZ, Mohsen ; SAFFAR, Mohammad H. ; SABOKROU, Mohammad ; FATHY, Mahmood ; KLETTE, Reinhard ; HUANG, Fay: **"STFCN: spatio-temporal FCN for semantic video segmentation"**. In *arXiv preprint arXiv:1608.05971* (2016)

[Feng et al. 2020] FENG, Di ; HAASE-SCHUETZ, Christian ; ROSENBAUM, Lars ; HERTLEIN, Heinz ; GLAESER, Claudius ; TIMM, Fabian ; WIESBECK, Werner ; DIETMAYER, Klaus: **"Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges"**. In *IEEE Transactions on Intelligent Transportation Systems* (2020)

[Fenton 1970] FENTON, Robert E.: **"Automatic vehicle guidance and control—A state of the art survey"**. In *IEEE Transactions on Vehicular Technology* 19 (1970), number 1, pages 153–161

[Fischler and Bolles 1981] FISCHLER, Martin A. ; BOLLES, Robert C.: **"Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography"**. In *Communications of the ACM* 24 (1981), number 6, pages 381–395

[Foret et al. 2020] FORET, Pierre ; KLEINER, Ariel ; MOBAHI, Hossein ; NEYSHABUR, Behnam: **"Sharpness-Aware Minimization for Efficiently Improving Generalization"**. In *arXiv preprint arXiv:2010.01412* (2020)

[Forsyth and Ponce 2002] FORSYTH, David A. ; PONCE, Jean: *Computer vision: a modern approach*. Prentice Hall Professional Technical Reference, 2002

[Fridman et al. 2016] FRIDMAN, Lex ; LANGHANS, Philipp ; LEE, Joonbum ; REIMER, Bryan: **"Driver gaze region estimation without use of eye movement"**. In *IEEE Intelligent Systems* 31 (2016), number 3, pages 49–56

[Frintrop 2006] FRINTROP, Simone: *VOCUS: A Visual Attention System for Object Detection and Goal-directed Search, vol. 3899 of Lecture Notes in Artificial Intelligence (LNAI)*. Springer, Berlin/Heidelberg, 2006

[Frintrop et al. 2015] FRINTROP, Simone ; WERNER, Thomas ; MARTIN GARCIA, German: **"Traditional saliency reloaded: A good old model in new shape"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pages 82–90

[Fuller 2011] FULLER, Ray: **"Driver control theory: From task difficulty homeostasis to risk allostasis"**. In *Handbook of traffic psychology*. Elsevier, 2011, pages 13–26

[Gaihua et al. 2021] GAIHUA, Wang ; TIANLUN, Zhang ; YINGYING, Dai ; JINHENG, Lin ; LEI, Cheng: **"A Serial-Parallel Self-Attention Network Joint With Multi-Scale Dilated Convolution"**. In *IEEE Access* 9 (2021), pages 71909–71919

[Gao et al. 2018] GAO, Fei ; WU, Lijun ; ZHAO, Li ; QIN, Tao ; CHENG, Xueqi ; LIU, Tie-Yan: **"Efficient sequence learning with group recurrent networks"**. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for*

Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), 2018, pages 799–808

[Gastal and Oliveira 2011] GASTAL, Eduardo S. ; OLIVEIRA, Manuel M.: **"Domain transform for edge-aware image and video processing"**. In *ACM Transactions on Graphics (ToG)* Volume 30 ACM (event), 2011, pages 69

[Geiger et al. 2015] GEIGER, Andreas ; LENZ, P ; STILLER, Christoph ; URTASUN, Raquel: **"The KITTI vision benchmark suite"**. In *URL http://www. cvlibs. net/datasets/kitti* (2015)

[Geiger et al. 2012] GEIGER, Andreas ; LENZ, Philip ; URTASUN, Raquel: **"Are we ready for autonomous driving? the kitti vision benchmark suite"**. In *2012 IEEE Conference on Computer Vision and Pattern Recognition* IEEE (event), 2012, pages 3354–3361

[Girshick 2015] GIRSHICK, Ross: **"Fast r-cnn"**. In *Proceedings of the IEEE international conference on computer vision*, 2015, pages 1440–1448

[Girshick et al. 2014] GIRSHICK, Ross ; DONAHUE, Jeff ; DARRELL, Trevor ; MALIK, Jitendra: **"Rich feature hierarchies for accurate object detection and semantic segmentation"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pages 580–587

[Goodfellow et al. 2016a] GOODFELLOW, Ian ; BENGIO, Yoshua ; COURVILLE, Aaron: *Deep Learning*. MIT Press, 2016. – http://www.deeplearningbook.org

[Goodfellow et al. 2016b] GOODFELLOW, Ian ; BENGIO, Yoshua ; COURVILLE, Aaron ; BENGIO, Yoshua: *Deep learning*. MIT press Cambridge, 2016

[Goodfellow et al. 2014a] GOODFELLOW, Ian ; POUGET-ABADIE, Jean ; MIRZA, Mehdi ; XU, Bing ; WARDE-FARLEY, David ; OZAIR, Sherjil ; COURVILLE, Aaron ; BENGIO, Yoshua: **"Generative adversarial nets"**. In *Advances in neural information processing systems*, 2014, pages 2672–2680

[Goodfellow et al. 2014b] GOODFELLOW, Ian ; POUGET-ABADIE, Jean ; MIRZA, Mehdi ; XU, Bing ; WARDE-FARLEY, David ; OZAIR, Sherjil ; COURVILLE, Aaron ; BENGIO, Yoshua: **"Generative adversarial nets"**. In *Advances in neural information processing systems*, 2014, pages 2672–2680

[Goodfellow et al. 2013] GOODFELLOW, Ian J. ; WARDE-FARLEY, David ; MIRZA, Mehdi ; COURVILLE, Aaron ; BENGIO, Yoshua: **"Maxout networks"**. In *arXiv preprint arXiv:1302.4389* (2013)

[Gould et al. 2009] GOULD, Stephen ; FULTON, Richard ; KOLLER, Daphne: **"Decomposing a scene into geometric and semantically consistent regions"**. In *Computer Vision, 2009 IEEE 12th International Conference on* IEEE (event), 2009, pages 1–8

[Graves et al. 2013] GRAVES, Alex ; MOHAMED, Abdel-rahman ; HINTON, Geoffrey: **"Speech recognition with deep recurrent neural networks"**. In *2013 IEEE international conference on acoustics, speech and signal processing* Ieee (event), 2013, pages 6645–6649

[Harel et al. 2007] HAREL, Jonathan ; KOCH, Christof ; PERONA, Pietro: **"Graph-based visual saliency"**. In *Advances in neural information processing systems*, 2007, pages 545–552

[Hariharan et al. 2011] HARIHARAN, Bharath ; ARBELÁEZ, Pablo ; BOURDEV, Lubomir ; MAJI, Subhransu ; MALIK, Jitendra: **"Semantic contours from inverse detectors"**. (2011)

[He et al. 2017a] HE, Kaiming ; GKIOXARI, Georgia ; DOLLÁR, Piotr ; GIRSHICK, Ross: **"Mask r-cnn"**. In *Proceedings of the IEEE international conference on computer vision*, 2017, pages 2961–2969

[He et al. 2016] HE, Kaiming ; ZHANG, Xiangyu ; REN, Shaoqing ; SUN, Jian: **"Deep residual learning for image recognition"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pages 770–778

[He et al. 2017b] HE, Yang ; CHIU, Wei-Chen ; KEUPER, Margret ; FRITZ, Mario ; CAMPUS, SI: **"STD2P: RGBD Semantic Segmentation Using Spatio-Temporal Data-Driven Pooling."**. In *CVPR*, 2017, pages 7158–7167

[Heusel et al. 2017] HEUSEL, Martin ; RAMSAUER, Hubert ; UNTERTHINER, Thomas ; NESSLER, Bernhard ; HOCHREITER, Sepp: **"Gans trained by a two time-scale update rule converge to a local nash equilibrium"**. In *Advances in Neural Information Processing Systems*, 2017, pages 6626–6637

[Hinton 2009] HINTON, Geoffrey E.: **"Deep belief networks"**. In *Scholarpedia* 4 (2009), number 5, pages 5947

[Hinton et al. 2006] HINTON, Geoffrey E. ; OSINDERO, Simon ; TEH, Yee-Whye: **"A fast learning algorithm for deep belief nets"**. In *Neural computation* 18 (2006), number 7, pages 1527–1554

[Hirschmuller 2008] HIRSCHMULLER, H.: **"Stereo Processing by Semiglobal Matching and Mutual Information"**. In *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30 (2008), number 2, pages 328–341. DOI: 10.1109/TPAMI.2007.1166

[Hirschmuller and Scharstein 2008] HIRSCHMULLER, Heiko ; SCHARSTEIN, Daniel: **"Evaluation of stereo matching costs on images with radiometric differences"**. In *IEEE transactions on pattern analysis and machine intelligence* 31 (2008), number 9, pages 1582–1599

[Hochreiter and Schmidhuber 1997] HOCHREITER, Sepp ; SCHMIDHUBER, Jürgen: **"Long short-term memory"**. In *Neural computation* 9 (1997), number 8, pages 1735–1780

[Hoffman et al. 2016] HOFFMAN, Judy ; WANG, Dequan ; YU, Fisher ; DARRELL, Trevor: **"Fcns in the wild: Pixel-level adversarial and constraint-based adaptation"**. In *arXiv preprint arXiv:1612.02649* (2016)

[Holliday et al. 2017] HOLLIDAY, Andrew ; BAREKATAIN, Mohammadamin ; LAURMAA, Johannes ; KANDASWAMY, Chetak ; PRENDINGER, Helmut: **"Speedup of deep learning ensembles for semantic segmentation using a model compression technique"**. In *Computer Vision and Image Understanding* 164 (2017), pages 16–26

[Holschneider et al. 1990] HOLSCHNEIDER, Matthias ; KRONLAND-MARTINET, Richard ; MORLET, Jean ; TCHAMITCHIAN, Ph: **"A real-time algorithm for signal analysis with the help of the wavelet transform"**. In *Wavelets*. Springer, 1990, pages 286–297

[Hong et al. 2015] HONG, Seunghoon ; NOH, Hyeonwoo ; HAN, Bohyung: **"Decoupled deep neural network for semi-supervised semantic segmentation"**. In *Advances in neural information processing systems*, 2015, pages 1495–1503

[Hong et al. 2021] HONG, Yuanduo ; PAN, Huihui ; SUN, Weichao ; JIA, Yisong ; OTHERS: **"Deep dual-resolution networks for real-time and accurate semantic segmentation of road scenes"**. In *arXiv preprint arXiv:2101.06085* (2021)

[Hosang et al. 2015] HOSANG, Jan ; BENENSON, Rodrigo ; DOLLÁR, Piotr ; SCHIELE, Bernt: **"What makes for effective detection proposals?"**. In *IEEE transactions on pattern analysis and machine intelligence* 38 (2015), number 4, pages 814–830

[Hou et al. 2011] HOU, Xiaodi ; HAREL, Jonathan ; KOCH, Christof: **"Image signature: Highlighting sparse salient regions"**. In *IEEE transactions on pattern analysis and machine intelligence* 34 (2011), number 1, pages 194–201

[Hou and Zhang 2007] HOU, Xiaodi ; ZHANG, Liqing: **"Saliency detection: A spectral residual approach"**. In *2007 IEEE Conference on Computer Vision and Pattern Recognition* Ieee (event), 2007, pages 1–8

[Howard et al. 2017] HOWARD, Andrew G. ; ZHU, Menglong ; CHEN, Bo ; KALENICHENKO, Dmitry ; WANG, Weijun ; WEYAND, Tobias ; ANDREETTO, Marco ;

ADAM, Hartwig: **"Mobilenets: Efficient convolutional neural networks for mobile vision applications"**. In *arXiv preprint arXiv:1704.04861* (2017)

[Hsu et al. 2019] HSU, Kuang-Jui ; LIN, Yen-Yu ; CHUANG, Yung-Yu: **"Weakly Supervised Salient Object Detection by Learning A Classifier-Driven Map Generator"**. In *IEEE Transactions on Image Processing* (2019)

[Hu et al. 2021a] HU, Jian-Fang ; SUN, Jiangxin ; LIN, Zihang ; LAI, Jian-Huang ; ZENG, Wenjun ; ZHENG, Wei-Shi: **"APANet: Auto-Path Aggregation for Future Instance Segmentation Prediction"**. In *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021)

[Hu et al. 2021b] HU, Jie ; CAO, Liujuan ; LU, Yao ; ZHANG, ShengChuan ; WANG, Yan ; LI, Ke ; HUANG, Feiyue ; SHAO, Ling ; JI, Rongrong: **"ISTR: End-to-End Instance Segmentation with Transformers"**. In *arXiv preprint arXiv:2105.00637* (2021)

[Huang et al. 2017] HUANG, Gao ; LIU, Zhuang ; VAN DER MAATEN, Laurens ; WEINBERGER, Kilian Q.: **"Densely connected convolutional networks"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pages 4700–4708

[Huang et al. 2018a] HUANG, Gao ; YUAN, Yang ; XU, Qiantong ; GUO, Chuan ; SUN, Yu ; WU, Felix ; WEINBERGER, Kilian: **"An empirical study on evaluation metrics of generative adversarial networks"**. (2018)

[Huang et al. 2018b] HUANG, Xinyu ; CHENG, Xinjing ; GENG, Qichuan ; CAO, Binbin ; ZHOU, Dingfu ; WANG, Peng ; LIN, Yuanqing ; YANG, Ruigang: **"The ApolloScape Dataset for Autonomous Driving"**. In *arXiv preprint arXiv:1803.06184* (2018)

[Huang et al. 2015] HUANG, Xun ; SHEN, Chengyao ; BOIX, Xavier ; ZHAO, Qi: **"Salicon: Reducing the semantic gap in saliency prediction by adapting deep neural networks"**. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pages 262–270

[Huang et al. 2018c] HUANG, Zilong ; WANG, Xinggang ; WANG, Jiasi ; LIU, Wenyu ; WANG, Jingdong: **"Weakly-Supervised Semantic Segmentation Network with Deep Seeded Region Growing"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pages 7014–7023

[Hung et al. 2018] HUNG, Wei-Chih ; TSAI, Yi-Hsuan ; LIOU, Yan-Ting ; LIN, Yen-Yu ; YANG, Ming-Hsuan: **"Adversarial Learning for Semi-Supervised Semantic Segmentation"**. In *arXiv preprint arXiv:1802.07934* (2018)

[Iandola et al. 2016] IANDOLA, Forrest N. ; HAN, Song ; MOSKEWICZ, Matthew W. ; ASHRAF, Khalid ; DALLY, William J. ; KEUTZER, Kurt: **"SqueezeNet: AlexNet-level**

accuracy with 50x fewer parameters and¡ 0.5 MB model size". In *arXiv preprint arXiv:1602.07360* (2016)

[Ilg et al. 2017] ILG, Eddy ; MAYER, Nikolaus ; SAIKIA, Tonmoy ; KEUPER, Margret ; DOSOVITSKIY, Alexey ; BROX, Thomas: **"Flownet 2.0: Evolution of optical flow estimation with deep networks"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pages 2462–2470

[international 2016] INTERNATIONAL, SAE: **"Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles"**. In *SAE International,(J3016)* (2016)

[Ioffe and Szegedy 2015] IOFFE, Sergey ; SZEGEDY, Christian: **"Batch normalization: Accelerating deep network training by reducing internal covariate shift"**. In *International conference on machine learning* PMLR (event), 2015, pages 448–456

[Islam et al. 2017] ISLAM, Md A. ; NAHA, Shujon ; ROCHAN, Mrigank ; BRUCE, Neil ; WANG, Yang: **"Label refinement network for coarse-to-fine semantic segmentation"**. In *arXiv preprint arXiv:1703.00551* (2017)

[Isola et al. 2017] ISOLA, Phillip ; ZHU, Jun-Yan ; ZHOU, Tinghui ; EFROS, Alexei A.: **"Image-to-image translation with conditional adversarial networks"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pages 1125–1134

[Itti et al. 1998] ITTI, Laurent ; KOCH, Christof ; NIEBUR, Ernst: **"A model of saliency-based visual attention for rapid scene analysis"**. In *IEEE Transactions on Pattern Analysis & Machine Intelligence* (1998), number 11, pages 1254–1259

[Jaderberg et al. 2014] JADERBERG, Max ; VEDALDI, Andrea ; ZISSERMAN, Andrew: **"Speeding up convolutional neural networks with low rank expansions"**. In *arXiv preprint arXiv:1405.3866* (2014)

[Jain et al. 1995] JAIN, Ramesh ; KASTURI, Rangachar ; SCHUNCK, Brian G.: *Machine vision*. Volume 5. McGraw-hill New York, 1995

[Jain and Grauman 2014] JAIN, Suyog D. ; GRAUMAN, Kristen: **"Supervoxel-consistent foreground propagation in video"**. In *European Conference on Computer Vision* Springer (event), 2014, pages 656–671

[Jampani et al. 2016] JAMPANI, Varun ; KIEFEL, Martin ; GEHLER, Peter V.: **"Learning sparse high dimensional filters: Image filtering, dense crfs and bilateral neural networks"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pages 4452–4461

[Jégou et al. 2017] JÉGOU, Simon ; DROZDZAL, Michal ; VAZQUEZ, David ; ROMERO, Adriana ; BENGIO, Yoshua: **"The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pages 11–19

[Ji et al. 2019] JI, Xu ; HENRIQUES, Joao F. ; VEDALDI, Andrea: **"Invariant information clustering for unsupervised image classification and segmentation"**. In *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pages 9865–9874

[Jia et al. 2016] JIA, Cong ; QI, Jinqing ; LI, Xiaohui ; LU, Huchuan: **"Saliency detection via a unified generative and discriminative model"**. In *Neurocomputing* 173 (2016), pages 406–417

[Jiang et al. 2021] JIANG, Dingchao ; QU, Hua ; ZHAO, Jihong ; ZHAO, Jianlong ; LIANG, Wei: **"Multi-level graph convolutional recurrent neural network for semantic image segmentation"**. In *Telecommunication Systems* (2021), pages 1–14

[Jiang et al. 2017] JIANG, Jindong ; ZHANG, Zhijun ; HUANG, Yongqian ; ZHENG, Lunan: **"Incorporating depth into both CNN and CRF for indoor semantic segmentation"**. In *Software Engineering and Service Science (ICSESS), 2017 8th IEEE International Conference on* IEEE (event), 2017, pages 525–530

[Jiang et al. 2015] JIANG, Ming ; HUANG, Shengsheng ; DUAN, Juanyong ; ZHAO, Qi: **"SALICON: Saliency in Context"**. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015

[John et al. 2015] JOHN, Vijay ; YONEDA, Keisuke ; LIU, Zheng ; MITA, Seiichi: **"Saliency map generation by the convolutional neural network for real-time traffic light detection using template matching"**. In *IEEE transactions on computational imaging* 1 (2015), number 3, pages 159–173

[Jung et al. 2020] JUNG, Sukwoo ; CHO, Youngmok ; KIM, Doojun ; CHANG, Minho: **"Moving object detection from moving camera image sequences using an inertial measurement unit sensor"**. In *Applied Sciences* 10 (2020), number 1, pages 268

[Kemker et al. 2018] KEMKER, Ronald ; SALVAGGIO, Carl ; KANAN, Christopher: **"Algorithms for semantic segmentation of multispectral remote sensing imagery using deep learning"**. In *ISPRS journal of photogrammetry and remote sensing* 145 (2018), pages 60–77

[Khoreva et al. 2017] KHOREVA, Anna ; BENENSON, Rodrigo ; HOSANG, Jan H. ; HEIN, Matthias ; SCHIELE, Bernt: **"Simple Does It: Weakly Supervised Instance and Semantic Segmentation."**. In *CVPR* Volume 1, 2017, pages 3

[Kim and Milanfar 2013] KIM, Chelhwon ; MILANFAR, Peyman: **"Visual saliency in noisy images"**. In *Journal of vision* 13 (2013), number 4, pages 5–5

[Kim et al. 2016] KIM, Jihun ; KIM, Seonggyu ; MALLIPEDDI, Rammohan ; JANG, Giljin ; LEE, Minho: **"Adaptive driver assistance system based on traffic information saliency map"**. In *2016 International Joint Conference on Neural Networks (IJCNN)* IEEE (event), 2016, pages 1918–1923

[Kim et al. 2015] KIM, Yong-Deok ; PARK, Eunhyeok ; YOO, Sungjoo ; CHOI, Taelim ; YANG, Lu ; SHIN, Dongjun: **"Compression of deep convolutional neural networks for fast and low power mobile applications"**. In *arXiv preprint arXiv:1511.06530* (2015)

[Koch and Ullman 1987] KOCH, Christof ; ULLMAN, Shimon: **"Shifts in selective visual attention: towards the underlying neural circuitry"**. In *Matters of intelligence*. Springer, 1987, pages 115–141

[Koppula et al. 2011] KOPPULA, Hema S. ; ANAND, Abhishek ; JOACHIMS, Thorsten ; SAXENA, Ashutosh: **"Semantic labeling of 3d point clouds for indoor scenes"**. In *Advances in neural information processing systems*, 2011, pages 244–252

[Koutnik et al. 2014] KOUTNIK, Jan ; GREFF, Klaus ; GOMEZ, Faustino ; SCHMIDHUBER, Juergen: **"A clockwork rnn"**. In *arXiv preprint arXiv:1402.3511* (2014)

[Krähenbühl and Koltun 2011] KRÄHENBÜHL, Philipp ; KOLTUN, Vladlen: **"Efficient inference in fully connected crfs with gaussian edge potentials"**. In *Advances in neural information processing systems*, 2011, pages 109–117

[Krizhevsky and Hinton 2009] KRIZHEVSKY, Alex ; HINTON, Geoffrey: **"Learning multiple layers of features from tiny images"** / Citeseer. 2009. – Research Report

[Krizhevsky et al. 2012] KRIZHEVSKY, Alex ; SUTSKEVER, Ilya ; HINTON, Geoffrey E.: **"Imagenet classification with deep convolutional neural networks"**. In *Advances in neural information processing systems* 25 (2012), pages 1097–1105

[Kruthiventi et al. 2017] KRUTHIVENTI, Srinivas S. ; AYUSH, Kumar ; BABU, R V.: **"Deepfix: A fully convolutional neural network for predicting human eye fixations"**. In *IEEE Transactions on Image Processing* 26 (2017), number 9, pages 4446–4456

[Kuang et al. 2017] KUANG, Hulin ; YANG, Kai-Fu ; CHEN, Long ; LI, Yong-Jie ; CHAN, Leanne Lai H. ; YAN, Hong: **"Bayes saliency-based object proposal generator for nighttime traffic images"**. In *IEEE Transactions on Intelligent Transportation Systems* 19 (2017), number 3, pages 814–825

[Kuen et al. 2016] KUEN, Jason ; WANG, Zhenhua ; WANG, Gang: **"Recurrent attentional networks for saliency detection"**. In *Proceedings of the IEEE Conference on computer Vision and Pattern Recognition*, 2016, pages 3668–3677

[Kümmerer et al. 2014] KÜMMERER, Matthias ; THEIS, Lucas ; BETHGE, Matthias: **"Deep gaze i: Boosting saliency prediction with feature maps trained on imagenet"**. In *arXiv preprint arXiv:1411.1045* (2014)

[Kummerer et al. 2017] KUMMERER, Matthias ; WALLIS, Thomas S. ; GATYS, Leon A. ; BETHGE, Matthias: **"Understanding low-and high-level contributions to fixation prediction"**. In *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pages 4789–4798

[Kümmerle et al. 2015] KÜMMERLE, Rainer ; RUHNKE, Michael ; STEDER, Bastian ; STACHNISS, Cyrill ; BURGARD, Wolfram: **"Autonomous robot navigation in highly populated pedestrian zones"**. In *Journal of Field Robotics* 32 (2015), number 4, pages 565–589

[Kundu et al. 2009] KUNDU, Abhijit ; KRISHNA, K M. ; SIVASWAMY, Jayanthi: **"Moving object detection by multi-view geometric techniques from a single camera mounted robot"**. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems* IEEE (event), 2009, pages 4306–4312

[Kundu et al. 2016] KUNDU, Abhijit ; VINEET, Vibhav ; KOLTUN, Vladlen: **"Feature space optimization for semantic video segmentation"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pages 3168–3175

[Lai et al. 2011] LAI, Kevin ; BO, Liefeng ; REN, Xiaofeng ; FOX, Dieter: **"A large-scale hierarchical multi-view rgb-d object dataset"**. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on* IEEE (event), 2011, pages 1817–1824

[Lateef and Ruichek 2019] LATEEF, Fahad ; RUICHEK, Yassine: **"Survey on semantic segmentation using deep learning techniques"**. In *Neurocomputing* 338 (2019), pages 321–348

[LeCun et al. 1998] LECUN, Yann ; BOTTOU, Léon ; BENGIO, Yoshua ; HAFFNER, Patrick: **"Gradient-based learning applied to document recognition"**. In *Proceedings of the IEEE* 86 (1998), number 11, pages 2278–2324

[Lee et al. 2020] LEE, Hojoon ; YOON, Jeongsik ; JEONG, Yonghwan ; YI, Kyongsu: **"Moving Object Detection and Tracking Based on Interaction of Static Obstacle Map and Geometric Model-Free Approach for Urban Autonomous Driving"**. In *IEEE Transactions on Intelligent Transportation Systems* (2020)

[Lee et al. 2008] LEE, Sungkil ; KIM, Gerard J. ; CHOI, Seungmoon: **"Real-time tracking of visually attended objects in virtual environments and its application to LOD"**. In *IEEE Transactions on Visualization and Computer Graphics* 15 (2008), number 1, pages 6–19

[Lee and Park 2020] LEE, Youngwan ; PARK, Jongyoul: **"Centermask: Real-time anchor-free instance segmentation"**. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pages 13906–13915

[Lenz et al. 2011] LENZ, Philip ; ZIEGLER, Julius ; GEIGER, Andreas ; ROSER, Martin: **"Sparse scene flow segmentation for moving object detection in urban environments"**. In *2011 IEEE Intelligent Vehicles Symposium (IV)* IEEE (event), 2011, pages 926–932

[Leutenegger et al. 2011] LEUTENEGGER, Stefan ; CHLI, Margarita ; SIEGWART, Roland Y.: **"BRISK: Binary robust invariant scalable keypoints"**. In *2011 International conference on computer vision* Ieee (event), 2011, pages 2548–2555

[Li et al. 2013] LI, Fuxin ; KIM, Taeyoung ; HUMAYUN, Ahmad ; TSAI, David ; REHG, James M.: **"Video segmentation by tracking many figure-ground segments"**. In *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pages 2192–2199

[Li et al. 2016a] LI, Hao ; KADAV, Asim ; DURDANOVIC, Igor ; SAMET, Hanan ; GRAF, Hans P.: **"Pruning Filters for Efficient ConvNets"**. In *CoRR* abs/1608.08710 (2016)

[Li et al. 2016b] LI, Jun ; MEI, Xue ; PROKHOROV, Danil ; TAO, Dacheng: **"Deep neural network for structural prediction and lane detection in traffic scene"**. In *IEEE transactions on neural networks and learning systems* 28 (2016), number 3, pages 690–703

[Li et al. 2020a] LI, Tiantian ; ZHANG, Mengxi ; QI, Wenyuan ; ASMA, Evren ; QI, Jinyi: **"Motion correction of respiratory-gated PET images using deep learning based image registration framework"**. In *Physics in Medicine & Biology* 65 (2020), number 15, pages 155003

[Li et al. 2020b] LI, Xiangtai ; LI, Xia ; ZHANG, Li ; CHENG, Guangliang ; SHI, Jianping ; LIN, Zhouchen ; TAN, Shaohua ; TONG, Yunhai: **"Improving semantic segmentation via decoupled body and edge supervision"**. In *arXiv preprint arXiv:2007.10035* (2020)

[Li et al. 2020c] LI, Yanfen ; WANG, Hanxiang ; DANG, L M. ; NGUYEN, Tan N. ; HAN, Dongil ; LEE, Ahyun ; JANG, Insung ; MOON, Hyeonjoon: **"A deep learning-based hybrid framework for object detection and recognition in autonomous driving"**. In *IEEE Access* 8 (2020), pages 194228–194239

[Li et al. 2020d] LI, Yanwei ; SONG, Lin ; CHEN, Yukang ; LI, Zeming ; ZHANG, Xiangyu ; WANG, Xingang ; SUN, Jian: **"Learning dynamic routing for semantic segmentation"**. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pages 8553–8562

[Li et al. 2018] LI, Yule ; SHI, Jianping ; LIN, Dahua: **"Low-Latency Video Semantic Segmentation"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pages 5997–6005

[Li et al. 2016c] LI, Zhen ; GAN, Yukang ; LIANG, Xiaodan ; YU, Yizhou ; CHENG, Hui ; LIN, Liang: **"Lstm-cf: Unifying context modeling and fusion with lstms for rgb-d scene labeling"**. In *European Conference on Computer Vision* Springer (event), 2016, pages 541–557

[Liang et al. 2018] LIANG, Xiaodan ; ZHOU, Hongfei ; XING, Eric: **"Dynamic-structured Semantic Propagation Network"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pages 752–761

[Lin et al. 2017a] LIN, Di ; CHEN, Guangyong ; COHEN-OR, Daniel ; HENG, Pheng-Ann ; HUANG, Hui: **"Cascaded feature network for semantic segmentation of rgb-d images"**. In *Computer Vision (ICCV), 2017 IEEE International Conference on* IEEE (event), 2017, pages 1320–1328

[Lin et al. 2017b] LIN, Guosheng ; MILAN, Anton ; SHEN, Chunhua ; REID, Ian: **"Refinenet: Multi-path refinement networks for high-resolution semantic segmentation"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pages 1925–1934

[Lin et al. 2015] LIN, Guosheng ; SHEN, Chunhua ; REID, Ian ; HENGEL, Anton van den: **"Deeply Learning the Messages in Message Passing Inference"**. In CORTES, C. (editors) ; LAWRENCE, N. D. (editors) ; LEE, D. D. (editors) ; SUGIYAMA, M. (editors) ; GARNETT, R. (editors): *Advances in Neural Information Processing Systems 28*. Curran Associates, Inc., 2015, pages 361–369

[Lin et al. 2016] LIN, Guosheng ; SHEN, Chunhua ; VAN DEN HENGEL, Anton ; REID, Ian: **"Efficient piecewise training of deep structured models for semantic segmentation"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pages 3194–3203

[Lin et al. 2013] LIN, Min ; CHEN, Qiang ; YAN, Shuicheng: **"Network in network"**. In *arXiv preprint arXiv:1312.4400* (2013)

[Lin and Wang 2014] LIN, Tsung-Han ; WANG, Chieh-Chih: **"Deep learning of spatio-temporal features with geometric-based moving point detection for motion seg-**

mentation". In *2014 IEEE International Conference on Robotics and Automation (ICRA)* IEEE (event), 2014, pages 3058–3065

[Lin et al. 2017c] LIN, Tsung-Yi ; DOLLÁR, Piotr ; GIRSHICK, Ross ; HE, Kaiming ; HARIHARAN, Bharath ; BELONGIE, Serge: **"Feature pyramid networks for object detection"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pages 2117–2125

[Lin et al. 2014a] LIN, Tsung-Yi ; MAIRE, Michael ; BELONGIE, Serge ; HAYS, James ; PERONA, Pietro ; RAMANAN, Deva ; DOLLÁR, Piotr ; ZITNICK, C L.: **"Microsoft coco: Common objects in context"**. In *European conference on computer vision* Springer (event), 2014, pages 740–755

[Lin et al. 2014b] LIN, Tsung-Yi ; MAIRE, Michael ; BELONGIE, Serge ; HAYS, James ; PERONA, Pietro ; RAMANAN, Deva ; DOLLÁR, Piotr ; ZITNICK, C L.: **"Microsoft coco: Common objects in context"**. In *European conference on computer vision* Springer (event), 2014, pages 740–755

[Liu et al. 2011] LIU, Ce ; YUEN, Jenny ; TORRALBA, Antonio: **"Nonparametric scene parsing via label transfer"**. In *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33 (2011), number 12, pages 2368–2382

[Liu et al. 2015a] LIU, Ming-Yu ; LIN, Shuoxin ; RAMALINGAM, Srikumar ; TUZEL, Oncel: **"Layered interpretation of street view images"**. In *arXiv preprint arXiv:1506.04723* (2015)

[Liu and Han 2018] LIU, Nian ; HAN, Junwei: **"A deep spatial contextual long-term recurrent convolutional network for saliency detection"**. In *IEEE Transactions on Image Processing* 27 (2018), number 7, pages 3264–3274

[Liu et al. 2018] LIU, Shu ; QI, Lu ; QIN, Haifang ; SHI, Jianping ; JIA, Jiaya: **"Path aggregation network for instance segmentation"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pages 8759–8768

[Liu et al. 2016] LIU, Shu ; QI, Xiaojuan ; SHI, Jianping ; ZHANG, Hong ; JIA, Jiaya: **"Multi-scale patch aggregation (mpa) for simultaneous detection and segmentation"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pages 3141–3149

[Liu et al. 2019] LIU, Wei ; LIAO, Shengcai ; REN, Weiqiang ; HU, Weidong ; YU, Yinan: **"High-level semantic feature detection: A new perspective for pedestrian detection"**. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pages 5187–5196

[Liu et al. 2015b] LIU, Ziwei ; LI, Xiaoxiao ; LUO, Ping ; LOY, Chen-Change ; TANG, Xiaoou: **"Semantic image segmentation via deep parsing network"**. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pages 1377–1385

[Long et al. 2015] LONG, Jonathan ; SHELHAMER, Evan ; DARRELL, Trevor: **"Fully convolutional networks for semantic segmentation"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pages 3431–3440

[Lovász et al. 1993] LOVÁSZ, László ; OTHERS: **"Random walks on graphs: A survey"**. In *Combinatorics, Paul erdos is eighty* 2 (1993), number 1, pages 1–46

[Luo et al. 2017] LUO, Ping ; WANG, Guangrun ; LIN, Liang ; WANG, Xiaogang: **"Deep dual learning for semantic image segmentation"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA*, 2017, pages 21–26

[Luo et al. 2021] LUO, Wenfeng ; YANG, Meng ; ZHENG, Weishi: **"Weakly-supervised semantic segmentation with saliency and incremental supervision updating"**. In *Pattern Recognition* 115 (2021), pages 107858

[Lyu et al. 2020] LYU, Nengchao ; WEN, Jiaqiang ; DUAN, Zhicheng ; WU, Chaozhong: **"Vehicle Trajectory Prediction and Cut-In Collision Warning Model in a Connected Vehicle Environment"**. In *IEEE Transactions on Intelligent Transportation Systems* (2020)

[Ma et al. 2018] MA, Ningning ; ZHANG, Xiangyu ; ZHENG, Hai-Tao ; SUN, Jian: **"Shufflenet v2: Practical guidelines for efficient cnn architecture design"**. In *arXiv preprint arXiv:1807.11164* 1 (2018)

[Mamalet and Garcia 2012] MAMALET, Franck ; GARCIA, Christophe: **"Simplifying convnets for fast learning"**. In *International Conference on Artificial Neural Networks* Springer (event), 2012, pages 58–65

[Maron and Lozano-Pérez 1998] MARON, Oded ; LOZANO-PÉREZ, Tomás: **"A framework for multiple-instance learning"**. In *Advances in neural information processing systems*, 1998, pages 570–576

[Marszalek and Schmid 2007] MARSZALEK, Marcin ; SCHMID, Cordelia: **"Accurate object localization with shape masks"**. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on* IEEE (event), 2007, pages 1–8

[Martinsson and Mogren 2019] MARTINSSON, John ; MOGREN, Olof: **"Semantic segmentation of fashion images using feature pyramid networks"**. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, 2019, pages 0–0

[Mehta et al. 2018] MEHTA, Sachin ; RASTEGARI, Mohammad ; CASPI, Anat ; SHAPIRO, Linda ; HAJISHIRZI, Hannaneh: **"Espnet: Efficient spatial pyramid of dilated convolutions for semantic segmentation"**. In *Proceedings of the european conference on computer vision (ECCV)*, 2018, pages 552–568

[Menze and Geiger 2015] MENZE, Moritz ; GEIGER, Andreas: **"Object scene flow for autonomous vehicles"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pages 3061–3070

[Milioto et al. 2018] MILIOTO, Andres ; LOTTES, Philipp ; STACHNISS, Cyrill: **"Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in CNNs"**. In *2018 IEEE international conference on robotics and automation (ICRA)* IEEE (event), 2018, pages 2229–2235

[Mirza and Osindero 2014] MIRZA, Mehdi ; OSINDERO, Simon: **"Conditional generative adversarial nets"**. In *arXiv preprint arXiv:1411.1784* (2014)

[Montabone and Soto 2010] MONTABONE, Sebastian ; SOTO, Alvaro: **"Human detection using a mobile platform and novel features derived from a visual saliency mechanism"**. In *Image and Vision Computing* 28 (2010), number 3, pages 391–402

[Mostajabi et al. 2015] MOSTAJABI, Mohammadreza ; YADOLLAHPOUR, Payman ; SHAKHNAROVICH, Gregory: **"Feedforward semantic segmentation with zoom-out features"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pages 3376–3385

[Mottaghi et al. 2014] MOTTAGHI, Roozbeh ; CHEN, Xianjie ; LIU, Xiaobai ; CHO, Nam-Gyu ; LEE, Seong-Whan ; FIDLER, Sanja ; URTASUN, Raquel ; YUILLE, Alan: **"The role of context for object detection and semantic segmentation in the wild"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pages 891–898

[Mukherjee et al. 2019] MUKHERJEE, Prerana ; SHARMA, Manoj ; MAKWANA, Megh ; SINGH, Ajay P. ; UPADHYAY, Avinash ; TRIVEDI, Akkshita ; LALL, Brejesh ; CHAUDHURY, Santanu: **"DSAL-GAN: Denoising based Saliency Prediction with Generative Adversarial Networks"**. In *arXiv preprint arXiv:1904.01215* (2019)

[Munoz-Organero et al. 2018] MUNOZ-ORGANERO, Mario ; RUIZ-BLAQUEZ, Ramona ; SÁNCHEZ-FERNÁNDEZ, Luis: **"Automatic detection of traffic lights, street crossings and urban roundabouts combining outlier detection and deep learning classification techniques based on GPS traces while driving"**. In *Computers, Environment and Urban Systems* 68 (2018), pages 1–8

[Nanfack et al. 2018] NANFACK, Geraldin ; ELHASSOUNY, Azeddine ; THAMI, Rachid Oulad H.: **"Squeeze-SegNet: a new fast deep convolutional neural network for semantic segmentation"**. In *Tenth International Conference on Machine Vision (ICMV 2017)* Volume 10696 International Society for Optics and Photonics (event), 2018, pages 106962O

[Neuhold et al. 2017] NEUHOLD, Gerhard ; OLLMANN, Tobias ; BULÒ, Samuel R. ; KONTSCHIEDER, Peter: **"The Mapillary Vistas Dataset for Semantic Understanding of Street Scenes."**. In *ICCV*, 2017, pages 5000–5009

[Nilsson and Sminchisescu 2016] NILSSON, David ; SMINCHISESCU, Cristian: **"Semantic video segmentation by gated recurrent flow propagation"**. In *arXiv preprint arXiv:1612.08871* 2 (2016)

[Noh et al. 2015] NOH, Hyeonwoo ; HONG, Seunghoon ; HAN, Bohyung: **"Learning deconvolution network for semantic segmentation"**. In *Proceedings of the IEEE international conference on computer vision*, 2015, pages 1520–1528

[O'Craven et al. 1999] O'CRAVEN, Kathleen M. ; DOWNING, Paul E. ; KANWISHER, Nancy: **"fMRI evidence for objects as the units of attentional selection"**. In *Nature* 401 (1999), number 6753, pages 584

[Oh et al. 2021] OH, Youngmin ; KIM, Beomjun ; HAM, Bumsub: **"Background-Aware Pooling and Noise-Aware Loss for Weakly-Supervised Semantic Segmentation"**. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pages 6913–6922

[Oršić andŠegvić 2021] ORŠIĆ, Marin ; ŠEGVIĆ, Siniša: **"Efficient semantic segmentation with pyramidal fusion"**. In *Pattern Recognition* 110 (2021), pages 107611

[Palazzi et al. 2018] PALAZZI, Andrea ; ABATI, Davide ; SOLERA, Francesco ; CUCCHIARA, Rita ; OTHERS: **"Predicting the Driver's Focus of Attention: the DR (eye) VE Project"**. In *IEEE transactions on pattern analysis and machine intelligence* 41 (2018), number 7, pages 1720–1733

[Pan and Jiang 2017] PAN, Hengyue ; JIANG, Hui: **"Supervised adversarial networks for image saliency detection"**. In *arXiv preprint arXiv:1704.07242* (2017)

[Pan et al. 2017] PAN, Junting ; FERRER, Cristian C. ; MCGUINNESS, Kevin ; O'CONNOR, Noel E. ; TORRES, Jordi ; SAYROL, Elisa ; NIETO, Xavier Giro-i: **"Salgan: Visual saliency prediction with generative adversarial networks"**. In *arXiv preprint arXiv:1701.01081* (2017)

[Papandreou et al. 2015] PAPANDREOU, George ; CHEN, Liang-Chieh ; MURPHY, Kevin P. ; YUILLE, Alan L.: **"Weakly-and semi-supervised learning of a deep convolutional network for semantic image segmentation"**. In *Proceedings of the IEEE international conference on computer vision*, 2015, pages 1742–1750

[Park et al. 2017] PARK, Seong-Jin ; HONG, Ki-Sang ; LEE, Seungyong: **"Rdfnet: Rgb-d multi-level residual feature fusion for indoor semantic segmentation"**. In *Proceedings of the IEEE international conference on computer vision*, 2017, pages 4980–4989

[Paszke et al. 2016] PASZKE, Adam ; CHAURASIA, Abhishek ; KIM, Sangpil ; CULURCIELLO, Eugenio: **"Enet: A deep neural network architecture for real-time semantic segmentation"**. In *arXiv preprint arXiv:1606.02147* (2016)

[Pathak et al. 2014] PATHAK, Deepak ; SHELHAMER, Evan ; LONG, Jonathan ; DARRELL, Trevor: **"Fully convolutional multi-class multiple instance learning"**. In *arXiv preprint arXiv:1412.7144* (2014)

[Peng et al. 2017] PENG, Chao ; ZHANG, Xiangyu ; YU, Gang ; LUO, Guiming ; SUN, Jian: **"Large kernel matters improve semantic segmentation by global convolutional network"**. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on* IEEE (event), 2017, pages 1743–1751

[Pinheiro and Collobert 2014] PINHEIRO, Pedro ; COLLOBERT, Ronan: **"Recurrent convolutional neural networks for scene labeling"**. In *International conference on machine learning* PMLR (event), 2014, pages 82–90

[Pinheiro and Collobert 2015] PINHEIRO, Pedro O. ; COLLOBERT, Ronan: **"From image-level to pixel-level labeling with convolutional networks"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pages 1713–1721

[Pohlen et al. 2017] POHLEN, Tobias ; HERMANS, Alexander ; MATHIAS, Markus ; LEIBE, Bastian: **"Full-resolution residual networks for semantic segmentation in street scenes"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pages 4151–4160

[Pont-Tuset et al. 2017] PONT-TUSET, Jordi ; PERAZZI, Federico ; CAELLES, Sergi ; ARBELÁEZ, Pablo ; SORKINE-HORNUNG, Alex ; VAN GOOL, Luc: **"The 2017 davis challenge on video object segmentation"**. In *arXiv preprint arXiv:1704.00675* (2017)

[Pylyshyn and Storm 1988] PYLYSHYN, Zenon W. ; STORM, Ron W.: **"Tracking multiple independent targets: Evidence for a parallel tracking mechanism"**. In *Spatial vision* 3 (1988), number 3, pages 179–197

[Qiu et al. 2018] QIU, Zhaofan ; YAO, Ting ; MEI, Tao: **"Learning deep spatio-temporal dependence for semantic video segmentation"**. In *IEEE Transactions on Multimedia* 20 (2018), number 4, pages 939–949

[Ra et al. 2018] RA, Moonsoo ; JUNG, Ho G. ; SUHR, Jae K. ; KIM, Whoi-Yul: **"Part-based vehicle detection in side-rectilinear images for blind-spot detection"**. In *Expert Systems with Applications* 101 (2018), pages 116–128

[Ramzy et al. 2019] RAMZY, Mohamed ; RASHED, Hazem ; SALLAB, Ahmad E. ; YOGAMANI, Senthil: **"RST-MODNet: Real-time Spatio-temporal Moving Object Detection for Autonomous Driving"**. In *arXiv preprint arXiv:1912.00438* (2019)

[Ranzato et al. 2007] RANZATO, Marc'Aurelio ; HUANG, Fu J. ; BOUREAU, Y-Lan ; LECUN, Yann: **"Unsupervised learning of invariant feature hierarchies with applications to object recognition"**. In *2007 IEEE conference on computer vision and pattern recognition* IEEE (event), 2007, pages 1–8

[Rashed et al. 2019] RASHED, Hazem ; RAMZY, Mohamed ; VAQUERO, Victor ; EL SALLAB, Ahmad ; SISTU, Ganesh ; YOGAMANI, Senthil: **"Fusemodnet: Real-time camera and lidar based moving object detection for robust low-light autonomous driving"**. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2019, pages 0–0

[Rasmussen 1983] RASMUSSEN, Jens: **"Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models"**. In *IEEE transactions on systems, man, and cybernetics* (1983), number 3, pages 257–266

[Rateke and von Wangenheim 2020] RATEKE, Thiago ; WANGENHEIM, Aldo von: **"Road obstacles positional and dynamic features extraction combining object detection, stereo disparity maps and optical flow data"**. In *Machine Vision and Applications* 31 (2020), number 7, pages 1–11

[Ren et al. 2015] REN, Shaoqing ; HE, Kaiming ; GIRSHICK, Ross ; SUN, Jian: **"Faster r-cnn: Towards real-time object detection with region proposal networks"**. In *arXiv preprint arXiv:1506.01497* (2015)

[Richter et al. 2016] RICHTER, Stephan R. ; VINEET, Vibhav ; ROTH, Stefan ; KOLTUN, Vladlen: **"Playing for data: Ground truth from computer games"**. In *European Conference on Computer Vision* Springer (event), 2016, pages 102–118

[Romera et al. 2017] ROMERA, Eduardo ; ALVAREZ, José M ; BERGASA, Luis M. ; ARROYO, Roberto: **"Erfnet: Efficient residual factorized convnet for real-time semantic segmentation"**. In *IEEE Transactions on Intelligent Transportation Systems* 19 (2017), number 1, pages 263–272

[Ronneberger et al. 2015] RONNEBERGER, Olaf ; FISCHER, Philipp ; BROX, Thomas: **"U-net: Convolutional networks for biomedical image segmentation"**. In *International Conference on Medical image computing and computer-assisted intervention* Springer (event), 2015, pages 234–241

[Ros et al. 2016a] ROS, German ; SELLART, Laura ; MATERZYNSKA, Joanna ; VAZQUEZ, David ; LOPEZ, Antonio M.: **"The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pages 3234–3243

[Ros et al. 2016b] ROS, German ; SELLART, Laura ; MATERZYNSKA, Joanna ; VAZQUEZ, David ; LOPEZ, Antonio M.: **"The SYNTHIA Dataset: A Large Collection of Synthetic Images for Semantic Segmentation of Urban Scenes"**. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016

[Rosenblatt 1961] ROSENBLATT, Frank: **"Principles of neurodynamics. perceptrons and the theory of brain mechanisms"** / Cornell Aeronautical Lab Inc Buffalo NY. 1961. – Research Report

[Russell et al. 2008] RUSSELL, Bryan C. ; TORRALBA, Antonio ; MURPHY, Kevin P. ; FREEMAN, William T.: **"LabelMe: a database and web-based tool for image annotation"**. In *International journal of computer vision* 77 (2008), number 1-3, pages 157–173

[Saito et al. 2017] SAITO, Shunta ; KEROLA, Tommi ; TSUTSUI, Satoshi: **"Superpixel clustering with deep features for unsupervised road segmentation"**. In *arXiv preprint arXiv:1711.05998* (2017)

[Saleh et al. 2018] SALEH, Fatemeh S. ; ALIAKBARIAN, Mohammad S. ; SALZMANN, Mathieu ; PETERSSON, Lars ; ALVAREZ, Jose M. ; GOULD, Stephen: **"Incorporating network built-in priors in weakly-supervised semantic segmentation"**. In *IEEE transactions on pattern analysis and machine intelligence* 40 (2018), number 6, pages 1382–1396

[Saleh et al. 2017] SALEH, Fatemehsadat ; AKBARIAN, Mohammad Sadegh A. ; SALZMANN, Mathieu ; PETERSSON, Lars ; ALVAREZ, Jose M.: **"Bringing Background into the Foreground: Making All Classes Equal in Weakly-Supervised Video Semantic Segmentation."**. In *ICCV*, 2017, pages 2125–2135

[Saleh et al. 2016] SALEH, Fatemehsadat ; ALIAKBARIAN, Mohammad S. ; SALZMANN, Mathieu ; PETERSSON, Lars ; GOULD, Stephen ; ALVAREZ, Jose M.: **"Built-in foreground/background prior for weakly-supervised semantic segmentation"**. In *European Conference on Computer Vision* Springer (event), 2016, pages 413–432

[Salimans et al. 2016] SALIMANS, Tim ; GOODFELLOW, Ian ; ZAREMBA, Wojciech ; CHEUNG, Vicki ; RADFORD, Alec ; CHEN, Xi: **"Improved techniques for training gans"**. In *Advances in neural information processing systems*, 2016, pages 2234–2242

[Salvador et al. 2017] SALVADOR, Amaia ; BELLVER, Miriam ; CAMPOS, Victor ; BARADAD, Manel ; MARQUES, Ferran ; TORRES, Jordi ; NIETO, Xavier Giro-i: **"Recurrent neural networks for semantic instance segmentation"**. In *arXiv preprint arXiv:1712.00617* (2017)

[Sankaranarayanan et al. 2018] SANKARANARAYANAN, Swami ; BALAJI, Yogesh ; JAIN, Arpit ; LIM, Ser N. ; CHELLAPPA, Rama: **"Learning from synthetic data: Addressing domain shift for semantic segmentation"**. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018

[Scharwächter et al. 2013] SCHARWÄCHTER, Timo ; ENZWEILER, Markus ; FRANKE, Uwe ; ROTH, Stefan: **"Efficient multi-cue scene segmentation"**. In *German Conference on Pattern Recognition* Springer (event), 2013, pages 435–445

[Sears and Pylyshyn 2000] SEARS, Christopher R. ; PYLYSHYN, Zenon W.: **"Multiple object tracking and attentional processing."**. In *Canadian Journal of Experimental Psychology/Revue canadienne de psychologie expérimentale* 54 (2000), number 1, pages 1

[Sengar and Mukhopadhyay 2020] SENGAR, Sandeep S. ; MUKHOPADHYAY, Susanta: **"Moving object detection using statistical background subtraction in wavelet compressed domain"**. In *Multimedia Tools and Applications* 79 (2020), number 9, pages 5919–5940

[Shelhamer et al. 2016] SHELHAMER, Evan ; RAKELLY, Kate ; HOFFMAN, Judy ; DARRELL, Trevor: **"Clockwork convnets for video semantic segmentation"**. In *European Conference on Computer Vision* Springer (event), 2016, pages 852–868

[Shen et al. 2017] SHEN, Falong ; GAN, Rui ; YAN, Shuicheng ; ZENG, Gang: **"Semantic segmentation via structured patch prediction, context crf and guidance crf"**. In *IEEE Conference on Computer Vision and Pattern Recognition* Volume 8, 2017

[Shi et al. 2015] SHI, Xingjian ; CHEN, Zhourong ; WANG, Hao ; YEUNG, Dit-Yan ; WONG, Wai-Kin ; WOO, Wang-chun: **"Convolutional LSTM network: A machine learning approach for precipitation nowcasting"**. In *arXiv preprint arXiv:1506.04214* (2015)

[Shotton et al. 2011a] SHOTTON, Jamie ; FITZGIBBON, Andrew ; COOK, Mat ; SHARP, Toby ; FINOCCHIO, Mark ; MOORE, Richard ; KIPMAN, Alex ; BLAKE, Andrew: **"Real-time human pose recognition in parts from single depth images"**. In *CVPR 2011* Ieee (event), 2011, pages 1297–1304

[Shotton et al. 2011b] SHOTTON, Jamie ; FITZGIBBON, Andrew ; COOK, Mat ; SHARP, Toby ; FINOCCHIO, Mark ; MOORE, Richard ; KIPMAN, Alex ; BLAKE, Andrew: **"Real-time human pose recognition in parts from single depth images"**. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on* IEEE (event), 2011, pages 1297–1304

[Shotton et al. 2008] SHOTTON, Jamie ; JOHNSON, Matthew ; CIPOLLA, Roberto: **"Semantic texton forests for image categorization and segmentation"**. In *2008 IEEE conference on computer vision and pattern recognition* IEEE (event), 2008, pages 1–8

[Shotton et al. 2006] SHOTTON, Jamie ; WINN, John ; ROTHER, Carsten ; CRIMINISI, Antonio: **"Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation"**. In *European conference on computer vision* Springer (event), 2006, pages 1–15

[Shuai et al. 2016] SHUAI, Bing ; ZUO, Zhen ; WANG, Bing ; WANG, Gang: **"Dag-recurrent neural networks for scene labeling"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pages 3620–3629

[Shuai et al. 2017] SHUAI, Bing ; ZUO, Zhen ; WANG, Bing ; WANG, Gang: **"Scene segmentation with dag-recurrent neural networks"**. In *IEEE transactions on pattern analysis and machine intelligence* 40 (2017), number 6, pages 1480–1493

[Si et al. 2019] SI, Haiyang ; ZHANG, Zhiqiang ; LV, Feifan ; YU, Gang ; LU, Feng: **"Real-time semantic segmentation via multiply spatial fusion network"**. In *arXiv preprint arXiv:1911.07217* (2019)

[Siam et al. 2018a] SIAM, Mennatullah ; EIKERDAWY, Sara ; GAMAL, Mostafa ; ABDEL-RAZEK, Moemen ; JAGERSAND, Martin ; ZHANG, Hong: **"Real-time segmentation with appearance, motion and geometry"**. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* IEEE (event), 2018, pages 5793–5800

[Siam et al. 2018b] SIAM, Mennatullah ; MAHGOUB, Heba ; ZAHRAN, Mohamed ; YOGAMANI, Senthil ; JAGERSAND, Martin ; EL-SALLAB, Ahmad: **"Modnet: Motion and appearance based moving object detection network for autonomous driving"**. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)* IEEE (event), 2018, pages 2859–2864

[Siam et al. 2017] SIAM, Mennatullah ; VALIPOUR, Sepehr ; JAGERSAND, Martin ; RAY, Nilanjan: **"Convolutional gated recurrent networks for video segmentation"**. In *Image Processing (ICIP), 2017 IEEE International Conference on* IEEE (event), 2017, pages 3090–3094

[Silberman et al. 2012] SILBERMAN, Nathan ; HOIEM, Derek ; KOHLI, Pushmeet ; FERGUS, Rob: **"Indoor segmentation and support inference from rgbd images"**. In *European Conference on Computer Vision* Springer (event), 2012, pages 746–760

[Simon et al. 2009] SIMON, Ludovic ; TAREL, Jean-Philippe ; BRÉMOND, Roland: **"Alerting the drivers about road signs with poor visual saliency"**. In *2009 IEEE Intelligent Vehicles Symposium* IEEE (event), 2009, pages 48–53

[Simonyan and Zisserman 2014] SIMONYAN, Karen ; ZISSERMAN, Andrew: **"Very deep convolutional networks for large-scale image recognition"**. In *arXiv preprint arXiv:1409.1556* (2014)

[Song et al. 2015] SONG, Shuran ; LICHTENBERG, Samuel P. ; XIAO, Jianxiong: **"Sun rgb-d: A rgb-d scene understanding benchmark suite"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pages 567–576

[Souly et al. 2017] SOULY, Nasim ; SPAMPINATO, Concetto ; SHAH, Mubarak: **"Semi and weakly supervised semantic segmentation using generative adversarial network"**. In *arXiv preprint arXiv:1703.09695* (2017)

[Strudel et al. 2021] STRUDEL, Robin ; GARCIA, Ricardo ; LAPTEV, Ivan ; SCHMID, Cordelia: **"Segmenter: Transformer for Semantic Segmentation"**. In *arXiv preprint arXiv:2105.05633* (2021)

[Sun et al. 2019] SUN, Ke ; ZHAO, Yang ; JIANG, Borui ; CHENG, Tianheng ; XIAO, Bin ; LIU, Dong ; MU, Yadong ; WANG, Xinggang ; LIU, Wenyu ; WANG, Jingdong: **"High-resolution representations for labeling pixels and regions"**. In *arXiv preprint arXiv:1904.04514* (2019)

[Szegedy et al. 2017] SZEGEDY, Christian ; IOFFE, Sergey ; VANHOUCKE, Vincent ; ALEMI, Alexander: **"Inception-v4, inception-resnet and the impact of residual connections on learning"**. In *Proceedings of the AAAI Conference on Artificial Intelligence* Volume 31, 2017

[Szegedy et al. 2015] SZEGEDY, Christian ; LIU, Wei ; JIA, Yangqing ; SERMANET, Pierre ; REED, Scott ; ANGUELOV, Dragomir ; ERHAN, Dumitru ; VANHOUCKE, Vincent ; RABINOVICH, Andrew: **"Going deeper with convolutions"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pages 1–9

[Szegedy et al. 2016] SZEGEDY, Christian ; VANHOUCKE, Vincent ; IOFFE, Sergey ; SHLENS, Jon ; WOJNA, Zbigniew: **"Rethinking the inception architecture for computer vision"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pages 2818–2826

[Taghanaki et al. 2021] TAGHANAKI, Saeid A. ; ABHISHEK, Kumar ; COHEN, Joseph P. ; COHEN-ADAD, Julien ; HAMARNEH, Ghassan: **"Deep semantic segmentation of natural and medical images: A review"**. In *Artificial Intelligence Review* 54 (2021), number 1, pages 137–178

[Tao et al. 2020] TAO, Andrew ; SAPRA, Karan ; CATANZARO, Bryan: **"Hierarchical multi-scale attention for semantic segmentation"**. In *arXiv preprint arXiv:2005.10821* (2020)

[Tao et al. 2018] TAO, Xian ; ZHANG, Dapeng ; MA, Wenzhi ; LIU, Xilong ; XU, De: **"Automatic Metallic Surface Defect Detection and Recognition with Convolutional Neural Networks"**. In *Applied Sciences* 8 (2018), number 9, pages 1575

[Tavakoli et al. 2017] TAVAKOLI, Hamed R. ; BORJI, Ali ; LAAKSONEN, Jorma ; RAHTU, Esa: **"Exploiting inter-image similarity and ensemble of extreme learners for fixation prediction using deep features"**. In *Neurocomputing* 244 (2017), pages 10–18

[Tawari et al. 2018] TAWARI, Ashish ; MALLELA, Praneeta ; MARTIN, Sujitha: **"Learning to Attend to Salient Targets in Driving Videos Using Fully Convolutional RNN"**. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)* IEEE (event), 2018, pages 3225–3232

[Teichmann and Cipolla 2018] TEICHMANN, Marvin T. ; CIPOLLA, Roberto: **"Convolutional CRFs for Semantic Segmentation"**. In *arXiv preprint arXiv:1805.04777* (2018)

[Tighe and Lazebnik 2010] TIGHE, Joseph ; LAZEBNIK, Svetlana: **"Superparsing: scalable nonparametric image parsing with superpixels"**. In *European conference on computer vision* Springer (event), 2010, pages 352–365

[Tran et al. 2015] TRAN, Du ; BOURDEV, Lubomir ; FERGUS, Rob ; TORRESANI, Lorenzo ; PALURI, Manohar: **"Learning spatiotemporal features with 3D convolutional networks"**. In *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pages 4489–4497

[Treisman and Gelade 1980] TREISMAN, Anne M. ; GELADE, Garry: **"A feature-integration theory of attention"**. In *Cognitive psychology* 12 (1980), number 1, pages 97–136

[Treml et al. 2016] TREML, Michael ; ARJONA-MEDINA, José ; UNTERTHINER, Thomas ; DURGESH, Rupesh ; FRIEDMANN, Felix ; SCHUBERTH, Peter ; MAYR, Andreas ; HEUSEL, Martin ; HOFMARCHER, Markus ; WIDRICH, Michael ; OTHERS: **"Speeding up semantic segmentation for autonomous driving"**. In *MLITS, NIPS Workshop* Volume 2, 2016

[Uijlings et al. 2013] UIJLINGS, Jasper R. ; VAN DE SANDE, Koen E. ; GEVERS, Theo ; SMEULDERS, Arnold W.: **"Selective search for object recognition"**. In *International journal of computer vision* 104 (2013), number 2, pages 154–171

[Underwood et al. 2011] UNDERWOOD, Geoffrey ; HUMPHREY, Katherine ; VAN LOON, Editha: **"Decisions about objects in real-world scenes are influenced by visual saliency before and during their inspection"**. In *Vision research* 51 (2011), number 18, pages 2031–2038

[Valada et al. 2019] VALADA, Abhinav ; MOHAN, Rohit ; BURGARD, Wolfram: **"Self-supervised model adaptation for multimodal semantic segmentation"**. In *International Journal of Computer Vision* (2019), pages 1–47

[Valada et al. 2016] VALADA, Abhinav ; OLIVEIRA, Gabriel L. ; BROX, Thomas ; BURGARD, Wolfram: **"Deep multispectral semantic scene understanding of forested environments using multimodal fusion"**. In *International Symposium on Experimental Robotics* Springer (event), 2016, pages 465–477

[Valada et al. 2017] VALADA, Abhinav ; VERTENS, Johan ; DHALL, Ankit ; BURGARD, Wolfram: **"Adapnet: Adaptive semantic segmentation in adverse environmental conditions"**. In *2017 IEEE International Conference on Robotics and Automation (ICRA)* IEEE (event), 2017, pages 4644–4651

[Vaswani et al. 2017] VASWANI, Ashish ; SHAZEER, Noam ; PARMAR, Niki ; USZKOREIT, Jakob ; JONES, Llion ; GOMEZ, Aidan N. ; KAISER, Łukasz ; POLOSUKHIN, Illia: **"Attention is all you need"**. In *Advances in neural information processing systems*, 2017, pages 5998–6008

[Vemulapalli et al. 2016] VEMULAPALLI, Raviteja ; TUZEL, Oncel ; LIU, Ming-Yu ; CHELLAPA, Rama: **"Gaussian conditional random field network for semantic segmentation"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pages 3224–3233

[Vertens et al. 2017] VERTENS, Johan ; VALADA, Abhinav ; BURGARD, Wolfram: **"Smsnet: Semantic motion segmentation using deep convolutional neural networks"**. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* IEEE (event), 2017, pages 582–589

[Vig et al. 2014] VIG, Eleonora ; DORR, Michael ; COX, David: **"Large-scale optimization of hierarchical features for saliency prediction in natural images"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pages 2798–2805

[Visin et al. 2015] VISIN, F ; KASTNER, K ; CHO, K ; MATTEUCCI, M ; COURVILLE, A ; BENGIO, Y: **"A recurrent neural network based alternative to convolutional networks"**. In *arXiv preprint arXiv:1505.00393* (2015)

[Visin et al. 2016] VISIN, Francesco ; CICCONE, Marco ; ROMERO, Adriana ; KASTNER, Kyle ; CHO, Kyunghyun ; BENGIO, Yoshua ; MATTEUCCI, Matteo ; COURVILLE, Aaron: **"Reseg: A recurrent neural network-based model for semantic segmentation"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pages 41–48

[Vo and Lee 2018] VO, Duc M. ; LEE, Sang-Woong: **"Semantic image segmentation using fully convolutional neural networks with multi-scale images and multi-scale dilated convolutions"**. In *Multimedia Tools and Applications* (2018), pages 1–19

[Wang et al. 2021a] WANG, Chenjie ; LI, Chengyuan ; LIU, Jun ; LUO, Bin ; SU, Xin ; WANG, Yajun ; GAO, Yan: **"U2-ONet: A Two-Level Nested Octave U-Structure Network with a Multi-Scale Attention Mechanism for Moving Object Segmentation"**. In *Remote Sensing* 13 (2021), number 1, pages 60

[Wang et al. 2018] WANG, Panqu ; CHEN, Pengfei ; YUAN, Ye ; LIU, Ding ; HUANG, Zehua ; HOU, Xiaodi ; COTTRELL, Garrison: **"Understanding convolution for semantic segmentation"**. In *2018 IEEE winter conference on applications of computer vision (WACV)* IEEE (event), 2018, pages 1451–1460

[Wang and Neumann 2018] WANG, Weiyue ; NEUMANN, Ulrich: **"Depth-aware CNN for RGB-D Segmentation"**. In *arXiv preprint arXiv:1803.06791* (2018)

[Wang et al. 2019] WANG, Wenguan ; ZHAO, Shuyang ; SHEN, Jianbing ; HOI, Steven C. ; BORJI, Ali: **"Salient Object Detection With Pyramid Attention and Salient Edges"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pages 1448–1457

[Wang et al. 2016] WANG, Yuhang ; LIU, Jing ; LI, Yong ; YAN, Junjie ; LU, Hanqing: **"Objectness-aware semantic segmentation"**. In *Proceedings of the 24th ACM international conference on Multimedia*, 2016, pages 307–311

[Wang et al. 2017] WANG, Yunhe ; XU, Chang ; XU, Chao ; TAO, Dacheng: **"Beyond filters: Compact feature map for portable deep model"**. In *International Conference on Machine Learning*, 2017, pages 3703–3711

[Wang et al. 2021b] WANG, Yuqing ; XU, Zhaoliang ; WANG, Xinlong ; SHEN, Chunhua ; CHENG, Baoshan ; SHEN, Hao ; XIA, Huaxia: **"End-to-end video instance segmentation with transformers"**. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pages 8741–8750

[Wang et al. 2004] WANG, Zhou ; BOVIK, Alan C. ; SHEIKH, Hamid R. ; SIMONCELLI, Eero P. ; OTHERS: **"Image quality assessment: from error visibility to structural similarity"**. In *IEEE transactions on image processing* 13 (2004), number 4, pages 600–612

[Wei et al. 2018] WEI, Yunchao ; XIAO, Huaxin ; SHI, Honghui ; JIE, Zequn ; FENG, Jiashi ; HUANG, Thomas S.: **"Revisiting Dilated Convolution: A Simple Approach for Weakly-and Semi-Supervised Semantic Segmentation"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pages 7268–7277

[Wu et al. 2020a] WU, Tong ; LEI, Zhenzhen ; LIN, Bingqian ; LI, Cuihua ; QU, Yanyun ; XIE, Yuan: **"Patch Proposal Network for Fast Semantic Segmentation of High-Resolution Images"**. In *Proceedings of the AAAI Conference on Artificial Intelligence* Volume 34, 2020, pages 12402–12409

[Wu et al. 2017] WU, Yan ; YANG, Tao ; ZHAO, Junqiao ; GUAN, Linting ; LI, Jiqian: **"Fully combined convolutional network with soft cost function for traffic scene parsing"**. In *International Conference on Intelligent Computing* Springer (event), 2017, pages 725–731

[Wu et al. 2020b] WU, Yue ; GAO, Rongrong ; PARK, Jaesik ; CHEN, Qifeng: **"Future video synthesis with object motion prediction"**. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pages 5539–5548

[Wu et al. 2019a] WU, Yuxin ; KIRILLOV, Alexander ; MASSA, Francisco ; LO, Wan-Yen ; GIRSHICK, Ross: *Detectron2*. https://github.com/facebookresearch/detectron2. 2019

[Wu et al. 2016a] WU, Zifeng ; SHEN, Chunhua ; HENGEL, Anton van d.: **"Bridging category-level and instance-level semantic image segmentation"**. In *arXiv preprint arXiv:1605.06885* (2016)

[Wu et al. 2016b] WU, Zifeng ; SHEN, Chunhua ; HENGEL, Anton van d.: **"High-performance semantic segmentation using very deep fully convolutional networks"**. In *arXiv preprint arXiv:1604.04339* (2016)

[Wu et al. 2019b] WU, Zifeng ; SHEN, Chunhua ; VAN DEN HENGEL, Anton: **"Wider or deeper: Revisiting the resnet model for visual recognition"**. In *Pattern Recognition* 90 (2019), pages 119–133

[Xia et al. 2018] XIA, Ye ; ZHANG, Danqing ; KIM, Jinkyu ; NAKAYAMA, Ken ; ZIPSER, Karl ; WHITNEY, David: **"Predicting driver attention in critical situations"**. In *Asian Conference on Computer Vision* Springer (event), 2018, pages 658–674

[Xiao et al. 2013] XIAO, Jianxiong ; OWENS, Andrew ; TORRALBA, Antonio: **"Sun3d: A database of big spaces reconstructed using sfm and object labels"**. In *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pages 1625–1632

[Xie et al. 2021] XIE, Enze ; WANG, Wenhai ; YU, Zhiding ; ANANDKUMAR, Anima ; ALVAREZ, Jose M. ; LUO, Ping: **"SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers"**. In *arXiv preprint arXiv:2105.15203* (2021)

[Xie et al. 2017] XIE, Saining ; GIRSHICK, Ross ; DOLLÁR, Piotr ; TU, Zhuowen ; HE, Kaiming: **"Aggregated residual transformations for deep neural networks"**. In *Computer Vision and Pattern Recognition (CVPR), 2017 IEEE Conference on* IEEE (event), 2017, pages 5987–5995

[Xu et al. 2021] XU, Lian ; XUE, Hao ; BENNAMOUN, Mohammed ; BOUSSAID, Farid ; SOHEL, Ferdous: **"Atrous convolutional feature network for weakly supervised semantic segmentation"**. In *Neurocomputing* 421 (2021), pages 115–126

[Yan et al. 2019] YAN, Zhi ; SUN, Li ; KRAJNIK, Tomas ; RUICHEK, Yassine: **"EU Long-term Dataset with Multiple Sensors for Autonomous Driving"**. In *CoRR* abs/1909.03330 (2019). – URL http://arxiv.org/abs/1909.03330

[Yan et al. 2020] YAN, Zhi ; SUN, Li ; KRAJNIK, Tomas ; RUICHEK, Yassine: **"EU Long-term Dataset with Multiple Sensors for Autonomous Driving"**. In *Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020

[Yang et al. 2021] YANG, Michael Y. ; KUMAAR, Saumya ; LYU, Ye ; NEX, Francesco: **"Real-time Semantic Segmentation with Context Aggregation Network"**. In *ISPRS Journal of Photogrammetry and Remote Sensing* 178 (2021), pages 124–134

[Yang et al. 2019a] YANG, Tao ; WU, Yan ; ZHAO, Junqiao ; GUAN, Linting: **"Semantic segmentation via highly fused convolutional network with multiple soft cost functions"**. In *Cognitive Systems Research* 53 (2019), pages 20–30

[Yang et al. 2019b] YANG, Zou ; ZHIDING, Yu ; XIAOFENG, Liu ; KUMAR, BVK ; JIN-SONG, Wang: **"Confidence regularized self-training"**. In *ICCV*, 2019

[Ye et al. 2018] YE, Linwei ; LIU, Zhi ; WANG, Yang: **"Learning Semantic Segmentation with Diverse Supervision"**. In *arXiv preprint arXiv:1802.00509* (2018)

[Yoo and Lee 2019] YOO, Jisang ; LEE, Gyu-cheol: **"Moving Object Detection Using an Object Motion Reflection Model of Motion Vectors"**. In *Symmetry* 11 (2019), number 1, pages 34

[Yousefhussien et al. 2018] YOUSEFHUSSIEN, Mohammed ; KELBE, David J. ; IENTILUCCI, Emmett J. ; SALVAGGIO, Carl: **"A multi-scale fully convolutional network for semantic labeling of 3D point clouds"**. In *ISPRS Journal of Photogrammetry and Remote Sensing* (2018)

[Yu and Koltun 2015] YU, Fisher ; KOLTUN, Vladlen: **"Multi-scale context aggregation by dilated convolutions"**. In *arXiv preprint arXiv:1511.07122* (2015)

[Yu et al. 2017] YU, Fisher ; KOLTUN, Vladlen ; FUNKHOUSER, Thomas: **"Dilated residual networks"**. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pages 472–480

[Yu et al. 2018] YU, Fisher ; XIAN, Wenqi ; CHEN, Yingying ; LIU, Fangchen ; LIAO, Mike ; MADHAVAN, Vashisht ; DARRELL, Trevor: **"Bdd100k: A diverse driving video database with scalable annotation tooling"**. In *arXiv preprint arXiv:1805.04687* (2018)

[Yu et al. 2019a] YU, Lingli ; XIA, Xumei ; ZHOU, Kaijun: **"Traffic sign detection based on visual co-saliency in complex scenes"**. In *Applied Intelligence* 49 (2019), number 2, pages 764–790

[Yu et al. 2019b] YU, Yang ; KURNIANGGORO, Laksono ; JO, Kang-Hyun: **"Moving object detection for a moving camera based on global motion compensation and adaptive background model"**. In *International Journal of Control, Automation and Systems* 17 (2019), number 7, pages 1866–1874

[Zablocki et al. 2021] ZABLOCKI, Éloi ; BEN-YOUNES, Hédi ; PÉREZ, Patrick ; CORD, Matthieu: **"Explainability of vision-based autonomous driving systems: Review and challenges"**. In *arXiv preprint arXiv:2101.05307* (2021)

[Zeiler and Fergus 2014] ZEILER, Matthew D. ; FERGUS, Rob: **"Visualizing and understanding convolutional networks"**. In *European conference on computer vision* Springer (event), 2014, pages 818–833

[Zhang et al. 2020] ZHANG, Jia ; LI, Zhixin ; ZHANG, Canlong ; MA, Huifang: **"Robust Semi-Supervised Semantic Segmentation Based on Self-Attention and Spectral Normalization"**. In *2020 International Joint Conference on Neural Networks (IJCNN)* IEEE (event), 2020, pages 1–8

[Zhang and Sclaroff 2013] ZHANG, Jianming ; SCLAROFF, Stan: **"Saliency detection: A boolean map approach"**. In *Proceedings of the IEEE international conference on computer vision*, 2013, pages 153–160

[Zhang and Zhou 2018] ZHANG, Lihe ; ZHOU, Qin: **"Salient object detection via proposal selection"**. In *Neurocomputing* 295 (2018), pages 59–71

[Zhang et al. 2018a] ZHANG, Rui ; LI, Guangyun ; LI, Minglei ; WANG, Li: **"Fusion of images and point clouds for the semantic segmentation of large-scale 3D scenes based on deep learning"**. In *ISPRS Journal of Photogrammetry and Remote Sensing* (2018)

[Zhang and Zhu 2020] ZHANG, Xun X. ; ZHU, Xu: **"Moving vehicle detection in aerial infrared image sequences via fast image registration and improved YOLOv3 network"**. In *International Journal of Remote Sensing* 41 (2020), number 11, pages 4312–4335

[Zhang et al. 2017] ZHANG, Yang ; DAVID, Philip ; GONG, Boqing: **"Curriculum domain adaptation for semantic segmentation of urban scenes"**. In *The IEEE International Conference on Computer Vision (ICCV)* Volume 2, 2017, pages 6

[Zhang and Chi 2020] ZHANG, Yunfeng ; CHI, Mingmin: **"Mask-R-FCN: A Deep Fusion Network for Semantic Segmentation"**. In *IEEE Access* 8 (2020), pages 155753–155765

[Zhang et al. 2018b] ZHANG, Yuxiao ; CHEN, Haiqiang ; HE, Yiran ; YE, Mao ; CAI, Xi ; ZHANG, Dan: **"Road segmentation for all-day outdoor robot navigation"**. In *Neurocomputing* 314 (2018), pages 316–325

[Zhao et al. 2018] ZHAO, Hengshuang ; QI, Xiaojuan ; SHEN, Xiaoyong ; SHI, Jianping ; JIA, Jiaya: **"Icnet for real-time semantic segmentation on high-resolution images"**. In *Proceedings of the European conference on computer vision (ECCV)*, 2018, pages 405–420

[Zhao et al. 2017] ZHAO, Hengshuang ; SHI, Jianping ; QI, Xiaojuan ; WANG, Xiaogang ; JIA, Jiaya: **"Pyramid scene parsing network"**. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017, pages 2881–2890

[Zhao et al. 2015] ZHAO, Rui ; OUYANG, Wanli ; LI, Hongsheng ; WANG, Xiaogang: **"Saliency detection by multi-context deep learning"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pages 1265–1274

[Zheng et al. 2015] ZHENG, Shuai ; JAYASUMANA, Sadeep ; ROMERA-PAREDES, Bernardino ; VINEET, Vibhav ; SU, Zhizhong ; DU, Dalong ; HUANG, Chang ; TORR, Philip H.: **"Conditional random fields as recurrent neural networks"**. In *Proceedings of the IEEE international conference on computer vision*, 2015, pages 1529–1537

[Zheng et al. 2021] ZHENG, Sixiao ; LU, Jiachen ; ZHAO, Hengshuang ; ZHU, Xiatian ; LUO, Zekun ; WANG, Yabiao ; FU, Yanwei ; FENG, Jianfeng ; XIANG, Tao ; TORR, Philip H. ; OTHERS: **"Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers"**. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pages 6881–6890

[Zhou et al. 2016a] ZHOU, Bolei ; KHOSLA, Aditya ; LAPEDRIZA, Agata ; OLIVA, Aude ; TORRALBA, Antonio: **"Learning deep features for discriminative localization"**. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pages 2921–2929

[Zhou et al. 2016b] ZHOU, Bolei ; ZHAO, Hang ; PUIG, Xavier ; FIDLER, Sanja ; BARRIUSO, Adela ; TORRALBA, Antonio: **"Semantic understanding of scenes through the ADE20K dataset"**. In *arXiv preprint arXiv:1608.05442* (2016)

[Zhou et al. 2017] ZHOU, Dingfu ; FRÉMONT, Vincent ; QUOST, Benjamin ; DAI, Yuchao ; LI, Hongdong: **"Moving object detection and segmentation in urban environments from a moving platform"**. In *Image and Vision Computing* 68 (2017), pages 76–87

[Zhu et al. 2018] ZHU, Dandan ; DAI, Lei ; LUO, Ye ; ZHANG, Guokai ; SHAO, Xuan ; ITTI, Laurent ; LU, Jianwei: **"Multi-scale adversarial feature learning for saliency detection"**. In *Symmetry* 10 (2018), number 10, pages 457

[Zhu et al. 2019] ZHU, Yi ; SAPRA, Karan ; REDA, Fitsum A. ; SHIH, Kevin J. ; NEWSAM, Shawn ; TAO, Andrew ; CATANZARO, Bryan: **"Improving semantic segmentation via video propagation and label relaxation"**. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pages 8856–8865

List of Figures

List of Tables

IV

APPENDIX

A

PUBLICATION

A.1/ JOURNALS

Lateef F, Ruichek Y. ”Survey on semantic segmentation using deep learning techniques. Neurocomputing”. 2019; 338:321-48.

Lateef F, Kas M, Ruichek Y. ”Saliency Heat-Map as Visual Attention for Autonomous Driving Using Generative Adversarial Network (GAN)”. IEEE Transactions on Intelligent Transportation Systems. 2021

Lateef F, Kas M, Ruichek Y. ”Semantic-Aware Object Identification in Urban Driving Scenarios”. Transportation Research Part C: Emerging Technologies. 2021 Dec (Under review)

A.2/ CONFERENCES

Lateef F, Kas M, Ruichek Y. ”Temporal Semantics Auto-encoding based Moving Objects Detection in Urban Driving Scenario”. IEEE Intelligent Vehicles Symposium. 2021

Document generated with LaTeX and:
the LaTeX style for PhD Thesis created by S. Galland — https://github.com/gallandarakhneorg/tex-templates
the tex-upmethodology package suite — http://www.arakhne.org/tex-upmethodology/

SPIM
école doctorale **sciences pour l'ingénieur et microtechniques**

Title: Semantic Analysis of the Driving Environment in Urban Scenarios

Keywords: Deep Learning, Autonomous Driving, Semantic Segmentation, Visual Attention, Generative Adversarial Network, Moving Object Detection, Motion Estimation, Motion Compensation.

Abstract:

Understanding urban scenes require recognizing the semantic constituents of a scene and the complex interactions between them. In this work, we explore and provide effective representations for understanding urban scenes based on in situ perception, which can be helpful for planning and decision-making in various complex urban environments and under a variety of environmental conditions. We first present a taxonomy of deep learning methods in the area of semantic segmentation, the most studied topic in the literature for understanding urban driving scenes. The methods are categorized based on their architectural structure and further elaborated with a discussion of their advantages, possible limitations, and future directions. Then, we proposed a new approach to visual attention for driving based on a conditional generative adversarial network. We have presented the well-known salience algorithms, both classical and Deep Learning approaches, used for visual attention. We built a large visual attention database on a new strategy for mining saliency heatmaps from existing driving datasets. We then proposed a novel object identification framework that combines motion and geometry cues to understand the urban driving environment. A new moving object detection model is developed by integrating an encoder-decoder network with semantic segmentation and a disparity estimator. An image registration algorithm is proposed along with the optical flow to compensate for ego-motion. Extensive empirical evaluations on various driving datasets show that all the proposed methods achieve remarkable performance in terms of accuracy and demonstrate the effectiveness of the essential techniques for scene understanding in autonomous driving.

Titre : Analyse Sémantique de l'Environnement de Conduite dans les Scénarios Urbains

Mots-clés : Apprentissage profond, Segmentation Sémantique, Attention Visuelle, Conduite autonome, Réseaux génératifs conditionnels (GAN), Détection d'objets en mouvement, Estimation de mouvement, Compensation de mouvement.

Résumé :

La tâche de compréhension des scènes urbaines nécessite la reconnaissance des constituants sémantiques de la scène et les interactions complexes entre eux. Par le biais de cette thèse, nous explorons et fournissons des représentations efficaces pour comprendre les scènes urbaines basées sur la perception, qui peuvent être utiles pour la planification et la prise de décision dans divers environnements urbains complexes et conditions environnementales variées. Nous présentons d'abord une taxonomie des méthodes d'apprentissage profond dans le domaine de la segmentation sémantique, en vue de l'intéret que porte la communauté scientifique à ce sujet pour la compréhension des scènes de conduite urbaine. Ainsi, nous avons d'abord classifié ces méthodes en fonction de leur structure architecturale afin d'élaborer ensuite une discussion sur leurs avantages, limites possibles et orientations futures. En suite, nous avons proposé une nouvelle approche de l'attention visuelle pour la conduite basée sur un réseau génératif conditionnel (GAN). Présentation des algorithmes de saillance bien connus, à la fois des approches classiques et d'apprentissage profond utilisées pour l'attention visuelle. Dans ce contexte, nous avons mis en place une large base de données d'attention visuelle basée sur une nouvelle stratégie d'extraction de cartes de saillance à partir d'un ensemble de données de conduite existant. Nous avons ensuite proposé un nouveau cadre d'identification d'objets qui combine des indices de mouvement et de géométrie pour comprendre l'environnement de conduite urbain. Par ailleurs, un nouveau modèle de détection d'objets en mouvement a été développé en intégrant un réseau codeur-décodeur couplé avec la segmentation sémantique et un réseau d'estimation de disparité. Un algorithme d'enregistrement d'image est proposé avec le flux optique pour compenser l'ego-mouvement. De nombreuses évaluations approfondies sur divers ensembles de données de conduite montrent que toutes les méthodes proposées atteignent des performances remarquables en termes de précision et démontrent l'efficacité des techniques essentielles pour la compréhension de la scène en conduite autonome.

UBFC UNIVERSITÉ BOURGOGNE FRANCHE-COMTÉ
Université Bourgogne Franche-Comté
32, avenue de l'Observatoire
25000 Besançon, France